



Hantek

DSO2D20 series
Digital oscilloscope

Programming Guide
202508

Warranties and Declarations

Copyright

The copyright of this document belongs to Qingdao Hantek Electronics Co., LTD.

Statement

Qingdao Hantek Electronics Co., Ltd. reserves the right to amend this document without prior notice. Qingdao Hantek Electronics Co., Ltd. promises that the information provided is correct and reliable but does not guarantee that this document is free from errors. Before using this product, please make sure that the specifications of relevant technical documents are the latest effective version. If you use documents or products of Qingdao Hantek Electronics Co., LTD and need products, patents or works of third parties to cooperate with them, you shall be responsible for obtaining the consent and authorization of the third parties. The above consent and authorization shall not be the liability of Hantek.

Product certification

Hantek certified DSO2D20 oscilloscope to meet China's national industry standards and has passed the CE certification.

Contact us

If you have any questions when using the products of Qingdao Hantek Electronics Co., LTD., you can obtain service and support through the following ways:

Email: service@hantek.com, support@hantek.com

Website: <http://www.hantek.com>

Contents

Contents	I
1 Document overview	1
2 SCPI Command Overview	3
3 Command System	6
3.1 General commands	6
3.1.1 *CLS	6
3.1.2 *IDN?	7
3.1.3 *RST	7
3.2 :CHANnel<n> Commands	8
3.2.1 :CHANnel<n>:BWLimit	8
3.2.2 :CHANnel<n>:COUPling	9
3.2.3 :CHANnel<n>: DISPlay	9
3.2.4 :CHANnel<n>: INVert	10
3.2.5 :CHANnel<n>: OFFSet	11
3.2.6 :CHANnel<n>: SCALe	11
3.2.7 :CHANnel<n>: VERNier	12
3.2.8 :CHANnel<n>: PROBe	13
3.2.9 :CHANnel<n>: UNITs	13

3.3	:TIMEbase Commands	14
3.3.1	:TIMEbase:WINDow:ENABle	14
3.3.2	:TIMEbase:WINDow:POSition	15
3.3.3	:TIMEbase:WINDow:SCALe	15
3.3.4	:TIMEbase:POSition.....	16
3.3.5	:TIMEbase:SCALe	17
3.3.6	:TIMEbase:RANGe	17
3.3.7	:TIMEbase:MODE	18
3.4	:ACQuire Commands	18
3.4.1	:ACQuire:POINts	18
3.4.2	:ACQuire:TYPE.....	19
3.4.3	:ACQuire:SRATe?	20
3.4.4	:ACQuire:COUNt	21
3.5	:TRIGger Commands	21
3.5.1	:TRIGger:MODE	21
3.5.2	:TRIGger:STATus?	22
3.5.3	:TRIGger:SWEep.....	22
3.5.4	:TRIGger:HOLDoff.....	23
3.5.5	:TRIGger:FORCe.....	24

3.5.6	:TRIGger:EDGE:SOURce	24
3.5.7	:TRIGger:EDGE:SLOPe.....	25
3.5.8	:TRIGger:EDGE:LEVel.....	25
3.5.9	:TRIGger:PULSe:SOURce	26
3.5.10	:TRIGger:PULSe:POLarity.....	27
3.5.11	:TRIGger:PULSe:WHEN.....	27
3.5.12	:TRIGger:PULSe:WIDTH	28
3.5.13	:TRIGger:PULSe:LEVel	29
3.5.14	:TRIGger:TV:SOURce	29
3.5.15	:TRIGger:TV:POLarity	30
3.5.16	:TRIGger:TV:MODE TV	30
3.5.17	:TRIGger:TV:LINE	31
3.5.18	:TRIGger:TV:STANdard.....	32
3.5.19	:TRIGger:SLOPe:SOURce	32
3.5.20	:TRIGger:SLOPe:POLarity	33
3.5.21	:TRIGger:SLOPe:WHEN	34
3.5.22	:TRIGger:SLOPe:WIDTH.....	34
3.5.23	:TRIGger:SLOPe:ALEVel	35
3.5.24	:TRIGger:SLOPe:BLEVel	36

3.5.25	:TRIGger:TIMEout:SOURce.....	36
3.5.26	:TRIGger:TIMEout:LEVel	37
3.5.27	:TRIGger:TIMEout:WIDth.....	37
3.5.28	:TRIGger:TIMEout:POLarity.....	38
3.5.29	:TRIGger:WINDOw:SOURce.....	39
3.5.30	:TRIGger:WINDOw:ALEVel	39
3.5.31	:TRIGger:WINDOw:BLEVel	40
3.5.32	:TRIGger:PATTern:SOURce	40
3.5.33	:TRIGger:PATTern:PATTern	41
3.5.34	:TRIGger:PATTern:LEVel.....	42
3.5.35	:TRIGger:INTERVal:SOURce	42
3.5.36	:TRIGger:INTERVal:SLOp	43
3.5.37	:TRIGger:INTERVal:WHEN.....	43
3.5.38	:TRIGger:INTERVal:TIME	44
3.5.39	:TRIGger:INTERVal:ALEVel	45
3.5.40	:TRIGger:UNDER_Am:SOURce	45
3.5.41	:TRIGger:UNDER_Am:POLarity	46
3.5.42	:TRIGger:UNDER_Am:WHEN.....	47
3.5.43	:TRIGger:UNDER_Am:TIME	47

3.5.44	:TRIGger:UNDER_Am:ALEVel.....	48
3.5.45	:TRIGger:UNDER_Am:BLEVel.....	49
3.5.46	:TRIGger:UART:SOURce	49
3.5.47	:TRIGger:UART:CONdition.....	50
3.5.48	:TRIGger:UART:BAUD	51
3.5.49	:TRIGger:UART:ALEVel	51
3.5.50	:TRIGger:UART:DATA.....	52
3.5.51	:TRIGger:UART:WIDTh	52
3.5.52	:TRIGger:UART:PARity	53
3.5.53	:TRIGger:LIN:SOURce	54
3.5.54	:TRIGger:LIN:IDLe.....	54
3.5.55	:TRIGger:LIN:BAUd	55
3.5.56	:TRIGger:LIN:CONdition.....	55
3.5.57	:TRIGger:LIN:ID.....	56
3.5.58	:TRIGger:LIN:DATA.....	57
3.5.59	:TRIGger:LIN:ALEVel	57
3.5.60	:TRIGger:CAN:SOURce	58
3.5.61	:TRIGger:CAN:IDLe.....	59
3.5.62	:TRIGger:CAN:BAUd	59

3.5.63	:TRIGger:CAN:CONdition.....	60
3.5.64	:TRIGger:CAN:ID.....	61
3.5.65	:TRIGger:CAN:DATA.....	61
3.5.66	:TRIGger:CAN:ALEVel	62
3.5.67	:TRIGger:SPI:SDA:SOURce.....	62
3.5.68	:TRIGger:SPI:SCL:SOURce	63
3.5.69	:TRIGger:SPI:SCK.....	64
3.5.70	:TRIGger:SPI:WIDTh	64
3.5.71	:TRIGger:SPI:DATA.....	65
3.5.72	:TRIGger:SPI:MASK	65
3.5.73	:TRIGger:SPI:ALEVel	66
3.5.74	:TRIGger:SPI:BLEVel	67
3.5.75	:TRIGger:IIC:SDA:SOURce.....	67
3.5.76	:TRIGger:IIC:SCL:SOURce	68
3.5.77	:TRIGger:IIC:CONditon.....	68
3.5.78	:TRIGger:IIC:ADDer.....	69
3.5.79	:TRIGger:IIC:DATA.....	70
3.5.80	:TRIGger:IIC:ALEVel	70
3.5.81	:TRIGger:IIC:BLEVel	71

3.6	:Calibration commands.....	72
3.6.1	:CALibrate:START	72
3.6.2	:CALibrate:STATus?	72
3.6.3	:CALibrate:QUIT	73
3.7	:MATH Commands.....	73
3.7.1	:MATH:DISPlay.....	73
3.7.2	:MATH:OPERator	74
3.7.3	:MATH:SOURce1	75
3.7.4	:MATH:SOURce2	75
3.7.5	:MATH:SCALE	76
3.7.6	:MATH:OFFSet	76
3.7.7	:MATH:FFT:SOURce	77
3.7.8	:MATH:FFT:WINDow	78
3.7.9	:MATH:FFT:UNIT	78
3.7.10	:MATH:FFT:HSCale.....	79
3.7.11	:MATH:FFT:HCENter.....	80
3.8	Waveform Data Command Subsystem	81
3.8.1	:WAVEform:DATA:ALL? <source>	81
3.9	:DISPlay Commands.....	82

3.9.1	:DISPlay:TYPE	82
3.9.2	:DISPlay:WBRightness	83
3.9.3	:DISPlay:GRID	84
3.9.4	:DISPlay:GBRightness	84
3.10	:CURSor Commands	85
3.10.1	:CURSor:MODE	85
3.10.2	:CURSor:MANual:TYPE	86
3.10.3	:CURSor:MANual:SOURce	86
3.10.4	:CURSor:MANual:AX	87
3.10.5	:CURSor:MANual:AXValue?	87
3.10.6	:CURSor:MANual:BX	88
3.10.7	:CURSor:MANual:BXValue?	89
3.10.8	:CURSor:MANual:AY	89
3.10.9	:CURSor:MANual:AYValue?	90
3.10.10	:CURSor:MANual:BY	90
3.10.11	:CURSor:MANual:BYValue?	91
3.10.12	:CURSor:TRACk:SOURceA	91
3.10.13	:CURSor:TRACk:SOURceB	92
3.10.14	:CURSor:TRACk:AX	93

3.10.15	:CURSor:TRACk:AXValue?	93
3.10.16	:CURSor:TRACk:AY	94
3.10.17	:CURSor:TRACk:AYValue?	94
3.10.18	:CURSor:TRACk:BX	95
3.10.19	:CURSor:TRACk:BXValue?	95
3.10.20	:CURSor:TRACk:BY	96
3.10.21	:CURSor:TRACk:BYValue?	97
3.11	:MEASure Commands.....	97
3.11.1	:MEASure:ENABle	97
3.11.2	:MEASure:SOURce	98
3.11.3	:MEASure:ADISplay	98
3.11.4	:MEASure:ITEM.....	99
3.11.5	:MEASure:GATE:ENABle	100
3.11.6	:MEASure:GATE:AY	100
3.11.7	:MEASure:GATE:BY	101
3.12	:MASK Commands.....	101
3.12.1	:MASK:ENABle	101
3.12.2	:MASK:SOURce	102
3.12.3	:MASK:MDISplay	103

3.12.4	:MASK:OUTPut.....	103
3.12.5	:MASK:SOOutput.....	104
3.12.6	:MASK:X	105
3.12.7	:MASK:Y	105
3.12.8	:MASK:CREate	106
3.13	:SYSTem Commands.....	107
3.13.1	:SYSTem:GAM?	107
3.13.2	:SYSTem:RAM?	107
3.13.3	:SYSTem:PON.....	108
3.13.4	:SYSTem:LANGuage.....	108
3.13.5	:SYSTem:LOCKed.....	109
3.14	[[:SOURce] Commands.....	109
3.14.1	:DDS:SWITCh	110
3.14.2	:DDS:TYPE	110
3.14.3	:DDS:FREQ	111
3.14.4	:DDS:AMP	111
3.14.5	:DDS:OFFSet.....	112
3.14.6	:DDS:DUTY	112
3.14.7	:DDS:WAVE:MODE.....	113

3.14.8	:DDS:MODE:TYPE	114
3.14.9	:DDS:MODE:WAVE:TYPE	114
3.14.10	:DDS:MODE:FREQ	115
3.14.11	:DDS:MODE:DEPTHordeviation	115
3.14.12	:DDS:BURSt:SWITCh	116
3.14.13	:DDS:BURSt:TYPE	116
3.14.14	:DDS:BURSt:CNT	117
3.14.15	:DDS:BURSt:TRIGger	118
3.14.16	:DDS:ARB:DAC16:BIN	118
3.15	:AUToscale	119
3.16	:RUNing	120
3.17	:SINGle	120
3.18	:BUS Commands	121
3.18.1	:BUS:CAN:SOURce	121
3.18.2	:BUS:CAN:THReshold	121
3.18.3	:BUS:CAN:STYPe	122
3.18.4	:BUS:CAN:BAUD	122
3.18.5	:BUS:CAN:SPOint	123
3.18.6	:BUS:IIC:SCLK:SOURce	124

3.18.7	:BUS:IIC:SDA:SOURce	124
3.18.8	:BUS:IIC:SCLK:THReshold	125
3.18.9	:BUS:IIC:SDA:THReshold	125
3.18.10	:BUS:IIC:ADDRes	126
3.18.11	:BUS:LIN:SOURce	127
3.18.12	:BUS:LIN:THReshold	127
3.18.13	:BUS:LIN:BAUD	128
3.18.14	:BUS:LIN:PARity	128
3.18.15	:BUS:LIN:STANdard	129
3.18.16	:BUS:RS232:BAUD	130
3.18.17	:BUS:RS232:TX	130
3.18.18	:BUS:RS232:TX:THReshold	131
3.18.19	:BUS:RS232:RX	132
3.18.20	:BUS:RS232:RX:THReshold	132
3.18.21	:BUS:RS232:POLarity	133
3.18.22	:BUS:RS232:ENDian	133
3.18.23	:BUS:RS232:DBITs	134
3.18.24	:BUS:RS232:PARity	134
3.18.25	:BUS:RS232:STOP	135

3.18.26	:BUS:SPI:SCLK:SOURce	136
3.18.27	:BUS:SPI:SCLK:SLOPe	136
3.18.28	:BUS:SPI:MISO:SOURce.....	137
3.18.29	:BUS:SPI:MOSI:SOURce.....	137
3.18.30	:BUS:SPI:SCLK:SOURce:THReshold	138
3.18.31	:BUS:SPI:MISO:SOURce:THReshold	138
3.18.32	:BUS:SPI:MOSI:SOURce:THReshold	139
3.18.33	:BUS:SPI:MODE	140
3.18.34	:BUS:SPI:CS:SOURce	140
3.18.35	:BUS:SPI:CS:POLarity	141
3.18.36	:BUS:SPI:CS:THReshold	141
3.18.37	:BUS:SPI:ENDian.....	142
3.18.38	:BUS:SPI:DBITs	143
3.18.39	:BUS:SPI:POLarity	143

1 Document overview

This document describes how to quickly understand the front and back panels, user interfaces, and basic operation methods of the DSO2D20 series digital oscilloscopes.



Tip:

The latest version of this manual can be downloaded at (<http://www.hantek.com>).

Document number:

202508

Software version:

Software upgrade may change or increase product functionalities, please pay attention to Hantek website for the latest version.

Document format conventions:

1 Virtual keys and main interface icons

Use **[name]** to represent virtual keys and main interface icons. For example, **[Utility]** is for .

2 Menu

Use "menu text (bold) + color" to represent a label or a menu option. For example, **Sound** means turn on or off sound of the machine operation.

3 Operation steps

Use "->" to represent the next step. For example, **[Utility]** -> **Language** means click **Utility** label before clicking **Language** menu.

Document content conventions:

DSO2D20 series tablet oscilloscope consists of the following models. Unless otherwise specified, this manual uses DSO2D50 as an example to describe the DSO2D20 series and basic operations.

Model	Channel	Sampling Rate	Bandwidth	Signal Source
DSO2C20	2	2GSa/s	200MHz	-
DSO2C35	2	2GSa/s	350MHz	-
DSO2C50	2	2GSa/s	500MHz	-

Model	Channel	Sampling Rate	Bandwidth	Signal Source
DSO2D20	2	2GSa/s	200MHz	1
DSO2D35	2	2GSa/s	350MHz	1
DSO2D50	2	2GSa/s	500MHz	1

2 SCPI Command Overview

SCPI (Standard Commands for Programmable Instruments) is a standardized instrument programming language that is built upon the existing standard IEEE 488.1 and IEEE 488.2 and conforms to various standards, such as the floating point operation rule in IEEE 754 standard, ISO 646 7-bit coded character set for information interchange (equivalent to ASCII programming). The SCPI commands provide a hierarchical tree structure, and consist of multiple subsystems. Each command subsystem consists of one root keyword and one or more sub-keywords.

Syntax

The command line usually starts with a colon; the keywords are separated by colons, and following the keywords are the parameter settings available. The command ending with a quotation mark indicates querying a certain function and returns the query results. The keywords of the command and the first parameter are separated by a space.

For example:

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

```
:ACQuire:TYPE?
```

ACQuire is the root keyword of the command, TYPE is the second-level keyword. The command line starts with a colon ":", and different levels of keywords are also separated by colons. <type> indicates a settable parameter. The command ending with a quotation mark "?" indicates querying a certain function. The command keywords :ACQuire:TYPE and the parameter <type> are separated by a space.

In some commands with parameters, ",", " is often used to separate multiple parameters.

For example:

```
[ :TRACe[<n>]]:DATA:VALue volatile, <points>, <data>
```

Symbol Description

The following symbols are not sent with the commands.

1. Braces { }

The contents in the braces can contain one or multiple parameters. These parameters can be omitted or used for several times. Parameters are usually separated by the vertical bar "|". When using the command, you must select one of the parameters.

2. Vertical Bar |

The vertical bar is used to separate multiple parameters. When using the command, you must select one of the parameters.

3. Square Brackets []

The contents in the square brackets can be omitted.

4. Angle Brackets < >

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

Parameter Type

1. Bool

The parameter can be set to ON, OFF, 1, or 0. For example:

:MEASure:ADISplay <bool>

:MEASure:ADISplay?

Wherein, <bool> can be set to {{1|ON}}{{0|OFF}}. The query returns 1 or 0.

2. Discrete

The parameter can 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 an abbreviated form: NORM, AVER, PEAK or HRES.

3. Integer

Unless otherwise specified, the parameter can be any integer (NR1 format) within the effective value range.



NOTE

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 an integer ranging from 0 to 100. The query returns an integer ranging from 0 to 100.

4. Real

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

:TRIGger:TIMEout:TIME <time>

:TRIGger:TIMEout:TIME?

Wherein, <time> can be set to any real number ranging from 1.6E-8 (that is, 16 ns) to 1E+1 (that is, 10 s). The query returns a real number in scientific notation.

5. ASCII String

The parameter can 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. They can all be in upper case or in lower case. If abbreviation is used, you must input all the capital letters in the command. For example:

:DISPlay:GBRightness?

can be abbreviated to

:DISP:GBR?

3 Command System

This chapter introduces the syntax, functions, parameters, and usage of each DSO2D20 command.

NOTE

1. Unless otherwise specified, the descriptions in this manual take DSO2D50 as an example to illustrate the commands of the DSO2D20 series.
2. For the parameter setting command (time, frequency, amplitude, etc.), the digital oscilloscope can only recognize the numbers, unable to recognize the unit sent together with them. The unit of the parameter is a default one. For the default units of various parameters, refer to the descriptions for the specified command.

3.1 General commands

General commands are used to query basic instrument information or perform commonly used basic operations. These commands usually start with "*" and the length of the command keyword is 3 characters.

3.1.1 *CLS

Syntax

*CLS

Description

Clear the values of all event registers to zero while clearing the error queue.

Parameter

N/A

Remarks

N/A

Return Format

N/A

Example

*CLS 16 /*Enable bit 4 (decimal 16) of the register.*/

3.1.2 *IDN?

Syntax

*IDN?

Description

Query the ID string of the instrument.

Parameter

N/A

Remarks

N/A

Return Format

Query returns HANTEK, <model>, <serial number>, <software version>.

Among them,

<model>: Instrument model;

<serial number>: instrument serial number;

<software version>: Instrument software version.

Example

*IDN? /*HANTEK, DSO2D50, CN2142000000035, (2023.03.25)*/

3.1.3 *RST

Syntax

*RST

Description

Restore the instrument to its factory default state.

Parameter

N/A

Remarks

N/A

Return Format

N/A

Example

```
*RST          /*Restore the instrument to its factory default state*/
```

3.2 :CHANnel<n> Commands

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

3.2.1 :CHANnel<n>:BWLimit

Syntax

```
:CHANnel<n>:BWLimit <type>
:CHANnel<n>:BWLimit?
```

Description

Set or query the bandwidth limit parameters for the specified channel.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1 2}	—
<type>	Discrete	{OFF 20M 100M 200M 350M}	OFF

Remarks

- OFF: Disable bandwidth limitation, allowing high-frequency components contained in the measured signal to pass through.
- 20M: Open the 20M bandwidth limit, and high-frequency components greater than 20M in the measured signal will be attenuated.
- 100M: Open the 100M bandwidth limit, and high-frequency components greater than 100M in the measured signal will be attenuated.
- 200M: Open the 200M bandwidth limit, and high-frequency components greater than 200M in the measured signal will be attenuated.
- 350M: Open the 350M bandwidth limit, and high-frequency components greater than 350M in the measured signal will be attenuated.

Return Format

Query returns OFF, 20M, 100M, 200M, 350M.

Example

```
:CHANnel1:BWLimit 100M          /* Turn on 100MHz bandwidth limit*/
:CHANnel1:BWLimit?              /*Query return 100M*/
```

3.2.2 :CHANnel<n>:COUPling

Syntax

```
:CHANnel<n>:COUPling<coupling>
:CHANnel<n>:COUPling?
```

Description

Sets or queries the coupling mode of the specified channel.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1 2}	—
<coupling>	Discrete	{AC DC GND}	DC

Remarks

AC: the DC components of the signal under test are blocked.

DC: both DC and AC components of the signal under test can pass through the channel.

GND: both DC and AC components of the signal under test are blocked.

Return Format

The query returns AC, DC, or GND.

Example

```
:CHANnel1:COUPling AC      /*Selects the AC coupling mode.*/
:CHANnel1:COUPling?        /*The query returns AC.*/
```

3.2.3 :CHANnel<n>: DISPlay

Syntax

```
:CHANnel<n>:DISPlay <bool>
:CHANnel<n>:DISPlay?
```

Description

Enables or disables the specified channel; or queries the on/off status of the specified channel.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1 2}	—

Name	Type	Range	Default
<bool>	Bool	{{1 ON}}{{0 OFF}}	CH1:1 ON CH2:1 OFF

Remarks

N/A

Return Format

The query returns 1 or 0.

Example

```
:CHANnel1:DISPlay ON          /*Enables CH1.*/
:CHANnel1:DISPlay?           /*The query returns 1.*/
```

3.2.4 :CHANnel<n>: INVert

Syntax

```
:CHANnel<n>:INVert <bool>
:CHANnel<n>:INVert?
```

Description

Turns on or off the waveform invert for the specified channel; or queries the on/off status of the waveform invert for the specified channel.

Parameter

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

Remarks

When the waveform invert is turned off, the waveform is displayed normally; when the waveform invert is turned on, the voltage values of the displayed waveform are inverted.

Return Format

The query returns 1 or 0.

Example

```
:CHANnel1:INVert ON          /*Enables the waveform invert for CH1.*/
:CHANnel1:INVert?           /*The query returns 1.*/
```

3.2.5 :CHANnel<n>: OFFSet

Syntax

:CHANnel<n>:OFFSet <offset>

:CHANnel<n>:OFFSet?

Description

Sets or queries the vertical offset of the specified channel. The default unit is V.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1 2}	--
<offset>	Real	When the input impedance is 1M Ω , $\pm 1V$ (500uV/div ~85mV/div) $\pm 1V$ (500uV/div ~100mV/div) $\pm 30V$ (200mV/div ~ 10V/div) When the input impedance is 50 Ω , $\pm 1V$ (1mV/div ~ 100mV/div) $\pm 4V$ (102mV/div ~ 1V/div)	0V (probe ratio 1X)

Remarks

The vertical displacement value set is influenced by the vertical gear and probe ratio. The range of legal values varies with the set vertical gear and probe ratio values. If you set a value that is offset beyond the legal value range, the offset value will automatically be set to the closest legal value.

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? /* Query returns 1.000e-02*/

3.2.6 :CHANnel<n>: SCALE

Syntax

:CHANnel<n>:SCALE <scale>

:CHANnel<n>:SCALE?

Description

Set or query the vertical gear of the specified channel, with a default unit of V.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1 2}	—
<scale>	Real	Related to probe ratio Probe ratio is 1X: 500uV to 10V	1V (probe ratio of 10X)

Remarks

The adjustable range of vertical gear is related to the current probe ratio.

Return Format

Query returns the vertical gear value in scientific counting form.

Example

```
:CHANnel1:SCALE 1          /* Set the vertical gear of CH1 to 1V */
:CHANnel1:SCALE?           /* Query returns 1.000e+00 */
```

3.2.7 :CHANnel<n>:VERNier

Syntax

```
:CHANnel<n>:VERNier <bool>
:CHANnel<n>:VERNier?
```

Description

Turn on or off the fine tuning function for the specified channel vertical gear, or query the status of the fine tuning function for the specified channel vertical gear.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1 2}	—
<bool>	Bool	{{1 ON}}{0 OFF}}	0 OFF

Remarks

The fine-tuning setting is turned off by default. At this time, you can only set the vertical gear in steps of 1-2 to 5, namely 500u, 1mV, 2mV, 5mV, 10mV... 10V (probe ratio is 1X). When fine-tuning settings are turned on, you can further adjust the vertical gear within a smaller range to improve vertical resolution. If the amplitude of the input waveform is slightly greater than the full scale in the current gear, and the amplitude displayed in the next gear waveform is slightly lower, fine tuning can be used to improve the waveform display amplitude, which is conducive to observing signal details.

Return Format

Query returns 1 or 0.

Example

```
:CHANnel1:VERNier ON      /* Turn on the fine adjustment function of CH1 vertical
gear */
:CHANnel1:VERNier?        /* Query returns 1*/
```

3.2.8 :CHANnel<n>:PROBe

Syntax

```
:CHANnel<n>:PROBe <atten>
:CHANnel<n>:PROBe?
```

Description

Set or query the probe ratio for the specified channel.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1 2}	—
<atten>	Discrete	{ X1 X10 X50 X100 X500 X1000}	1

Remarks

- Set the probe ratio to display the collected signal multiplied by a specified multiple (without affecting the actual amplitude of the signal).
- Set the probe ratio to affect the adjustable range of the current vertical gear.

Return Format

Query returns the probe attenuation ratio in scientific counting form.

Example

```
:CHANnel1:PROBe 10      /* Set the attenuation ratio of CH1 probe to 10X */
:CHANnel1:PROBe?        /* Query returns 1.000000 e+01*/
```

3.2.9 :CHANnel<n>:UNITs

Syntax

```
:CHANnel<n>:UNITs <atten>
:CHANnel<n>:UNITs?
```

Description

Set or query the display unit for a specified channel.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1 2}	—
<atten>	Discrete	{WATT AMPere VOLT U}	VOLT

Remarks

Set the units displayed for the current channel.

Return Format

Query the units displayed on the current channel.

Example

```
:CHANnel1:UNITs VOLT      /* Set the amplitude display unit of CH1 to V */
:CHANnel1:UNITs?          /* Query returns VOLT */
```

3.3 :TIMEbase Commands

The TIMEbase command is used to set the horizontal system, such as turning on delayed scanning, setting the horizontal time base mode, etc.

3.3.1 :TIMEbase:WINDow:ENABle

Syntax

```
:TIMEbase:WINDow:ENABle <bool>
:TIMEbase:WINDow:ENABle?
```

Description

Turn on or off the dual window function, or check the dual window status.

Parameter

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

Remarks

The dual window mode can be used to horizontally zoom in on a waveform for viewing waveform details.

Return Format

Query returns 1 or 0.

Example

```
:TImEbase:WINDow:ENABle ON      /* Turn on delayed scanning */
:TImEbase:WINDow:ENABle?        /* Query returns 1*/
```

3.3.2 :TImEbase:WINDow:POSition

Syntax

```
:TImEbase:WINDow:POSition <pos value>
:TImEbase:WINDow:POSition?
```

Description

Set or query the horizontal position of the main scanning zoom view.

Parameter

Name	Type	Range	Default
<pos value >	Bool	Value of horizontal position	--

Remarks

The main scanning range and main scanning horizontal position determine the range of this command, and the value of this command must keep the zoomed view window within the main scanning range.

Return Format

Return the position of the current horizontal window in the zoomed in view in scientific counting format.

Example

```
:TImEbase:WINDow:POSition 0.001    /* Set horizontal position 0.001s */
:TImEbase:WINDow:POSition?          /* Query returns 1*/
```

3.3.3 :TImEbase:WINDow:SCALe

Syntax

```
:TImEbase:WINDow:SCALe <scale>
:TImEbase:WINDow:SCALe?
```

Description

Set or query the main time base gear, with a default unit of s/div.

Parameter

Name	Type	Range	Default
<scale>	Integer	YT mode: 5ns/div to 1ks/div in steps of 1-2-5 Roll mode: 100ms/div to 50s/div in steps of 1-2-5	1μs/div

Remarks

When the horizontal time base mode is YT and the horizontal time base is 200ms/div or larger (i.e. "slow scan" mode), this command is not available during the stopping process of the oscilloscope.

Return Format

Query returns the main time base level in scientific counting form.

Example

```
:TIMEbase:MAIN:SCALE 0.0002      /* Set the main time base gear to 200 μ s/div */
:TIMEbase:MAIN:SCALE?             /* Query returned 2.000000e-04*/
```

3.3.4 :TIMEbase:POSition

Syntax

```
:TIMEbase:POSition <pos value>
:TIMEbase:POSition?
```

Description

Set or query the main time base offset.

Parameter

Name	Type	Range	Default
< pos value >	Integer	offset value	--

Remarks

N/A

Return Format

Return the main time base offset value in scientific counting format.

Example

```
:TIMEbase:POSition 0.0002      /* Set the main time base offset to 200us */
:TIMEbase:POSition?             /* Query returned 2.000000e-04*/
```

3.3.5 :TIMebase:SCALe

Syntax

:TIMebase:SCALe < scale value>

:TIMebase:SCALe?

Description

Set or query the horizontal scale (seconds/bars) of the main window.

Parameter

Name	Type	Range	Default
< scale value >	Integer	The current seconds per grid in the main window	--

Remarks

N/A

Return Format

Return the main window timing gear in scientific counting format.

Example

:TIMebase:SCALe 0.0002 /* Set the main time base offset to 200us */

:TIMebase:SCALe? /* Query returned 2.000000e-04*/

3.3.6 :TIMebase:RANGe

Syntax

:TIMebase:RANGe < range value>

:TIMebase:RANGe?

Description

Set or query the full-scale horizontal time of the main window.

Parameter

Name	Type	Range	Default
< range value >	Integer	range value	--

Remarks

N/A

Return Format

Return the full-scale time base range of the main window in scientific counting form.

Example

```
:TIMEbase:RANGe 0.0002      /* Set the main time base gear to 100us*/
:TIMEbase:RANGe?           /* Query returned 2.000000e-04*/
```

3.3.7 :TIMEbase:MODE

Syntax

```
:TIMEbase:MODE < value>
:TIMEbase:MODE?
```

Description

Set or query horizontal time base mode.

Parameter

Name	Type	Range	Default
< value >	Integer	MAIN XY ROLL	--

Remarks

- Main: YT mode
- XY: XY mode
- ROLL: ROLL mode

Return Format

Query returns MAIN, XY, or ROLL.

Example

```
:TIMEbase:MODE XY      /* Set the horizontal time base mode to XY */
:TIMEbase:MODE?        /* Query returned XY*/
```

3.4 :ACQuire Commands

The :ACQuire commands are used to set the memory depth of the oscilloscope, the acquisition mode, the average times, as well as query the current sample rate.

3.4.1 :ACQuire:POINTs

Syntax

```
:ACquire:POINts <Value>
```

```
:ACquire:POINts?
```

Description

Set or query storage depth.

Parameter

Name	Type	Range	Default
<Value>	Discrete	4K 40K 80K	4K

Remarks

N/A

Return Format

Query and return the numerical value of storage depth.

Example

```
:ACquire:POINts 4000          /* Set storage depth to 4k */
```

```
:ACquire:POINts?              /* Query returns 4000 */
```

3.4.2 :ACquire:TYPE

Syntax

```
:ACquire:TYPE <Value>
```

```
:ACquire:TYPE?
```

Description

Set or query the acquisition method for oscilloscope sampling.

Parameter

Name	Type	Range	Default
<Value>	Discrete	{NORMal AVERAges PEAK HRESolution}	NORMal

Remarks

- **NORMAL:** In this mode, the oscilloscope samples the signal at equal time intervals to reconstruct the waveform. For most waveforms, using this mode can produce the best display effect.
- **AVERAges:** In this mode, the oscilloscope averages multiple sampled waveforms to reduce random noise on the input signal and improve vertical resolution. The average number of times can be set by the: ACquire: AVERAges command. The higher the average frequency, the smaller the noise and the higher the vertical

resolution, but the displayed waveform also responds slower to waveform changes.

- **PEAK** (peak detection): In this mode, the oscilloscope collects the maximum and minimum values of the sampling interval signal to obtain the envelope of the signal or narrow pulses that may be lost. Using this mode can avoid signal aliasing, but the displayed noise is relatively high.
- **HRESolution** (High Resolution): This mode uses a supersampling technique to average adjacent points of the sampled waveform, reducing random noise on the input signal and producing a smoother waveform on the screen. Usually used when the sampling rate of a digital converter is higher than the storage rate of the acquisition memory.

Return Format

The query returns NORM, AVERAge, PEAK, or HRESolution.

Example

```
:ACQUIRE:TYPE AVERages          /* Set the acquisition method to average */
:ACQUIRE:TYPE?                  /* Query returns AVERAge */
```

3.4.3 :ACQUIRE:SRATE?

Syntax

```
:ACQUIRE:SRATE?
```

Description

Query the current sampling rate, with a default unit of Sa/s.

Parameter

N/A

Remarks

- The sampling rate refers to the frequency at which the oscilloscope samples the signal, which is the number of waveform points sampled per second.
- The relationship between sampling rate, storage depth, and waveform length satisfies the following equation:
- $\text{Storage depth} = \text{sampling rate} \times \text{Waveform length}$.

Return Format

The query returns the sampling rate in real form.

Example

```
:ACQUIRE:SRATE?                /* Query returned 1.250e+06*/
```

3.4.4 :ACQuire:COUNT

Syntax

:ACQuire:COUNT <Value>

:ACQuire:COUNT?

Description

Set or query the number of points that need to be averaged in average mode.

Parameter

Name	Type	Range	Default
<Value>	Discrete	4 8 16 32 64 128	4

Remarks

N/A

Return Format

Return the current average point count.

Example

:ACQuire:COUNT 64 /* Set average points to 64*/

:ACQuire:COUNT? /* Query returns 64 */

3.5 :TRIGger Commands

The TRIGGer command is used to set the triggering system of the oscilloscope.

3.5.1 :TRIGger:MODE

Syntax

:TRIGger:MODE <mode>

:TRIGger:MODE?

Description

Select or query the trigger type.

Parameter

Name	Type	Range	Default
<mode>	Discrete	<EDGE PULSe TV SLOPe TImeout WIN	EDGE

Name	Type	Range	Default
		dow PATtern INTerval UNDerthrow UART LIN CAN SPI IIC>	

Remarks

N/A

Return Format

Return EDGE, PULSe, TV, SLOPe, TIMEout, WINdow, PATtern, INTerval, UNDerthrow, UART, LIN, CAN, SPI, IIC.

Example

```
:TRIGger:MODE SLOPe          /* Set the trigger type to slope trigger */
:TRIGger:MODE?               /* Query returns SLOPE */
```

3.5.2 :TRIGger:STATus?

Syntax

:TRIGger:STATus?

Description

Query the current triggering status.

Parameter

N/A

Remarks

N/A

Return Format

Query returns TRIGed, NOTRIG.

Example

```
:TRIGger:STATus?           /* Query returns AUTO */
```

3.5.3 :TRIGger:SWEep

Syntax

```
:TRIGger:SWEep <sweep>
:TRIGger:SWEep?
```

Description

Set or query the triggering method.

Parameter

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

Remarks

- AUTO: Automatically triggered, with waveform display regardless of whether the triggering conditions are met.
- NORMAL: Normal trigger, displaying the waveform when the triggering conditions are met, maintaining the original waveform display when the triggering conditions are not met, and waiting for the next trigger.
- Single: Single trigger, the oscilloscope waits for triggering, displays the waveform when the triggering conditions are met, and then stops.

Return Format

Query returns AUTO NORM or SING.

Example

```
:TRIGger:SWEep SINGle      /* Choose a single trigger method */
:TRIGger:SWEep?            /* Query returns SING */
```

3.5.4 :TRIGger:HOLDoff

Syntax

```
:TRIGger:HOLDoff <value>
:TRIGger:HOLDoff?
```

Description

Set or query the trigger release time, with a default unit of seconds.

Parameter

Name	Type	Range	Default
<value>	Real	16ns-10s	1us

Remarks

Trigger holdoff can stabilize triggering of complex waveforms (such as pulse series). The holdoff time is the period during which the oscilloscope waits to re-enable the trigger circuit and will not trigger until the holdoff time has elapsed.

Return Format

Query to return the trigger suppression time in scientific notation.

Example

```
:TRIGger:HOLDoff 1      /* Set the trigger release time to 1ms */
:TRIGger:HOLDoff?      /* Query returns 1.000000 e-03*/
```

3.5.5 :TRIGger:FORCE**Syntax**

```
:TRIGger:FORCE
```

Description

Set the oscilloscope to forced trigger.

Parameter

N/A

Remarks

This command will make the oscilloscope capture the waveform even to meet the trigger condition.

Return Format

N/A

Example

```
:TRIGger:FORCE 1      /* Set the oscilloscope to forced trigger.*/
```

3.5.6 :TRIGger:EDGE:SOURce**Syntax**

```
:TRIGger:EDGE:SOURce <source>
:TRIGger:EDGE:SOURce?
```

Description

Set or query the edge-triggered trigger source.

Parameter

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

Remarks

When a digital channel is used as a trigger source channel, the digital probe must be connected to the oscilloscope.

Return Format

Query returns CHAN1, CHAN2.

Example

```
:TRIGger:EDGE:SOURce CHANnel1      /* Set the trigger source to CH1*/
:TRIGger:EDGE:SOURce?              /* Query returns CHAN1*/
```

3.5.7 :TRIGger:EDGE:SLOPe

Syntax

```
:TRIGger:EDGE:SLOPe <slope>
:TRIGger:EDGE:SLOPe?
```

Description

Set or query the edge type triggered by the edge.

Parameter

Name	Type	Range	Default
<slope>	Discrete	{RISIng FALLIng EITHer}	RISIng

Remarks

- RISIng: Rising edge
- FALLIng: Falling edge
- EITHer: Any edge

Return Format

Query returns RISI, FALL or EITH.

Example

```
:TRIGger:EDGE:SLOPe FALLIng      /* Set the edge type to the falling edge */
:TRIGger:EDGE:SLOPe?              /* Query returns FALL */
```

3.5.8 :TRIGger:EDGE:LEVeL

Syntax

```
:TRIGger:EDGE:LEVel <level>
```

```
:TRIGger:EDGE:LEVel?
```

Description

Set or query the triggering level when the edge is triggered, with the unit consistent with the current amplitude unit of the selected signal source.

Parameter

Name	Type	Range	Default
<level>	Real	(-5×VerticalScale-OFFSet)- (5× VerticalScale-OFFSet)	0

Remarks

This setting command is only valid when the selected signal source is an analog channel.

Return Format

Query to return the trigger level value in scientific counting form.

Example

```
:TRIGger:EDGE:LEVel 0.16      /* Set the trigger level to 160mV */
```

```
:TRIGger:EDGE:LEVel?          /* Query returns 1.600000 e-01*/
```

3.5.9 :TRIGger:PULSe:SOURce

Syntax

```
:TRIGger:PULSe:SOURce <source>
```

```
:TRIGger:PULSe:SOURce?
```

Description

Set or query the trigger source for pulse width triggering.

Parameter

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

Remarks

When a digital channel is used as a trigger source channel, the digital probe must be connected to the oscilloscope.

Return Format

Query returns CHAN1 CHAN2.

Example

```
:TRIGger:PULSe:SOURce CHANnel1      /* Set the trigger source to CH1*/
:TRIGger:PULSe:SOURce?              /* Query returns CHAN1*/
```

3.5.10 :TRIGger:PULSe:POLarity**Syntax**

```
:TRIGger:PULSe:POLarity <polarity>
:TRIGger:PULSe:POLarity?
```

Description

Set or query the triggering polarity of pulse width triggering.

Parameter

Name	Type	Range	Default
<polarity>	Discrete	{POSitive NEGActive}	POSitive

Remarks

- POSItive: Positive polarity
- NEGActive: Negative polarity

Return Format

Query returns POSItive, NEGActive.

Example

```
:TRIGger:PULSe:POLarity POSitive      /* Set the triggering polarity to positive
polarity */
:TRIGger:PULSe:POLarity?              /* Query returns POSItive */
```

3.5.11 :TRIGger:PULSe:WHEN**Syntax**

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

Description

Set or query the triggering conditions for pulse width triggering.

Parameter

Name	Type	Range	Default
<when>	Discrete	{EQUAL NEQUAL GREAT LESS }	EQUAL

Remarks

- Equal=(equal to time value): can only be triggered when the positive or negative pulse width of the input signal is equal to the set pulse width. [Pulse width error is 5%]
- NEQUAL= (Not equal to time value): Only when the positive or negative pulse width of the input signal is not equal to the set pulse width, can it trigger 【 Pulse width error is 5% 】.
- GREAT>(greater than time value): Only when the positive or negative pulse width of the input signal is greater than the set pulse width can it trigger [pulse width error is 5%].
- LESS<(less than time value): Only when the positive or negative pulse width of the input signal is less than the set pulse width can it trigger [pulse width error is 5%].

Return Format

Query returns QUAL, NEQUAL, GREAT, LESS.

Example

```
:TRIGger:PULSe:WHEN LESS      /* Set the trigger condition to LESS */
:TRIGger:PULSe:WHEN?          /* Query returns LESS */
```

3.5.12 :TRIGger:PULSe:WIDTh

Syntax

```
:TRIGger:PULSe:WIDTh <width>
:TRIGger:PULSe:WIDTh?
```

Description

Set or query the pulse width value when triggered, with a default unit of s.

Parameter

Name	Type	Range	Default
<width>	Real	8ns-10s	20ns

Remarks

This command is applicable to trigger conditions.

Return Format

Query returns the pulse width value in scientific counting form.

Example

```
:TRIGger:PULSe:WIDTh 0.000003      /* Set the pulse width value to 3 μ s */
:TRIGger:PULSe:WIDTh?              /* Query returned 3.000000 e-06*/
```

3.5.13 :TRIGger:PULSe:LEVel

Syntax

```
:TRIGger:PULSe:LEVel <level>
:TRIGger:PULSe:LEVel?
```

Description

Set or query the triggering level when pulse width is triggered, with the unit consistent with the current amplitude unit.

Parameter

Name	Type	Range	Default
<level>	Real	(-5 × VerticalScale - OFFSet)- (5 × VerticalScale - OFFSet)	0

Remarks

This setting command is only valid when the selected signal source is an analog channel.

Return Format

Query to return the trigger level value in scientific counting form.

Example

```
:TRIGger:PULSe:LEVel 0.16          /* Set the trigger level to 160mV */
:TRIGger:PULSe:LEVel?              /* Query returns 1.600000 e-01*/
```

3.5.14 :TRIGger:TV:SOURce

Syntax

```
:TRIGger:TV:SOURce <source>
:TRIGger:TV:SOURce?
```

Description

Set or query the trigger source of video triggers.

Parameter

Name	Type	Range	Default
------	------	-------	---------

Name	Type	Range	Default
<source>	discrete	{CHANnel1 CHANnel2}	CHANnel1

Remarks

N/A

Return Format

The query returns CHAN1, CHAN2.

Example

```
:TRIGger:TV:SOURce CHANnel2      /*Set the trigger source to CH2 */
:TRIGger:TV:SOURce?              /* Query returns CH2 */
```

3.5.15 :TRIGger:TV:POLarity

Syntax

```
:TRIGger:TV:POLarity <polarity>
:TRIGger:TV:POLarity?
```

Description

Select or query the TV polarity when triggered.

Parameter

Name	Type	Range	Default
<polarity>	Discrete	{POSitive NEGative}	POSitive

Remarks

N/A

Return Format

Query and return POS or NEG.

Example

```
:TRIGger:TV:POLarity POSitive      /* Set video polarity to positive polarity */
:TRIGger:TV:POLarity?              /* Query returns POS */
```

3.5.16 :TRIGger:TV:MODE TV

Syntax

```
:TRIGger:TV:MODE <mode>
:TRIGger:TV:MODE?
```

Description

Set or query the synchronization type when the video is triggered.

Parameter

Name	Type	Range	Default
<mode>	Discrete	{SCANLine LINE ODDField EVENfield ALINes}	SCANLine

Remarks

- SCANLine: Scanning line
- LINE: For NTSC and PAL/SECAM video standards, triggered on specified rows in odd or even fields.
- ODDField: Triggered at the rising edge of the first sawtooth wave in an odd numbered field.
- EVENField: Triggered at the rising edge of the first sawtooth wave in an even numbered field.
- ALINs: triggered on all horizontal synchronization pulses.

Return Format

Query returns SCANL LINE, ODDF, EVEN, or ALIN.

Example

```
:TRIGger:TV:MODE ODDField      /* Set synchronization type to odd field */
:TRIGger:TV:MODE?              /* Query returns ODDF */
```

3.5.17 :TRIGger:TV:LINE

Syntax

```
:TRIGger:TV:LINE <line>
:TRIGger:TV:LINE?
```

Description

Set or query the line number when the video trigger synchronization type is specified as a line.

Parameter

Name	Type	Range	Default
<line>	discrete	NTSC:1-525; PAL:1-625	1

Remarks

The line number of the specified line.

Return Format

Retrieve the row number of the specified row.

Example

```
:TRIGger:TV:LINE 100      /* Set line number to 100*/
:TRIGger:TV:LINE?        /* Query returns 100*/
```

3.5.18 :TRIGger:TV:STANdard

Syntax

```
:TRIGger:TV:STANdard <standard>
:TRIGger:TV:STANdard?
```

Description

Set or query the video standard triggered by video.

Parameter

Name	Type	Range	Default
<line>	discrete	NTSC;PAL	PAL

Remarks

N/A.

Return Format

Query returns NTSC, PAL.

Example

```
:TRIGger:TV:STANdard NTSC      /* Set the video standard to NTSC */
:TRIGger:TV:STANdard?          /* Query returns NTSC */
```

3.5.19 :TRIGger:SLOPe:SOURce

Syntax

```
:TRIGger:SLOPe:SOURce <source>
:TRIGger:SLOPe:SOURce?
```

Description

Set or query the trigger source for slope triggering.

Parameter

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

Remarks

Can only be triggered for analog channels.

Return Format

Query returns CHAN1 CHAN2.

Example

```
:TRIGger:SLOPe:SOURce CHANnel2      /* Set the trigger source to CH2*/
:TRIGger:SLOPe:SOURce?                /* Query returns CHAN2*/
```

3.5.20 :TRIGger:SLOPe:POLarity

Syntax

```
:TRIGger:SLOPe:POLarity <polarity>
:TRIGger:SLOPe:POLarity?
```

Description

Set or query the edge type triggered by slope.

Parameter

Name	Type	Range	Default
<polarity>	Discrete	{POSItive NEGAtive}	POSItive

Remarks

- POSItive: Rising edge triggered
- NEGAtive: Falling edge triggered

Return Format

Query returns POSItive, NEGAtive.

Example

```
:TRIGger:SLOPe:POLarity POSItive      /* Set rising edge trigger */
:TRIGger:SLOPe:POLarity?                /* Query returns POSItive */
```

3.5.21 :TRIGger:SLOPe:WHEN

Syntax

```
:TRIGger:SLOPe:WHEN <when>
```

```
:TRIGger:SLOPe:WHEN?
```

Description

Set or query the triggering conditions for slope triggering.

Parameter

Name	Type	Range	Default
<when>	Discrete	{ EQUAL NEQUAl GREAt LESS }	EQUAl

Remarks

- Equal=(equal to time value): can only be triggered when the positive or negative pulse width of the input signal is equal to the set pulse width. [Pulse width error is 5%]
- NEQUAl= (Not equal to time value): Only when the positive or negative pulse width of the input signal is not equal to the set pulse width, can it trigger 【 Pulse width error is 5% 】.
- GREAt>(greater than time value): Only when the positive or negative pulse width of the input signal is greater than the set pulse width can it trigger [pulse width error is 5%].
- LESS<(less than time value): Only when the positive or negative pulse width of the input signal is less than the set pulse width can it trigger [pulse width error is 5%].

Return Format

Query returns QUAL, NEQUAl, GREAt, LESS.

Example

```
:TRIGger:SLOPe:WHEN LESS      /* Set the trigger condition to LESS */
```

```
:TRIGger:SLOPe:WHEN?          /* Query returns LESS */
```

3.5.22 :TRIGger:SLOPe:WIDTH

Syntax

```
:TRIGger:SLOPe:WIDTH <time>
```

```
:TRIGger:SLOPe:WIDTH?
```

Description

Set or query the time value when the slope is triggered. The default unit is s.

Parameter

Name	Type	Range	Default
<time>	Real	8ns-10s	20ns

Remarks

This command is applicable to trigger conditions.

Return Format

Query returns a time value in scientific counting form.

Example

```
:TRIGger:SLOPe:WIDTH 0.000003      /* Set the time value to 3 μ s */
:TRIGger:SLOPe:WIDTH?                /* Query returned 3.000000 e-06*/
```

3.5.23 :TRIGger:SLOPe:ALEVel

Syntax

```
:TRIGger:SLOPe:ALEVel <level>
:TRIGger:SLOPe:ALEVel?
```

Description

Set or query the upper limit of the triggering level when the slope is triggered, with the unit consistent with the current amplitude unit.

Parameter

Name	Type	Range	Default
<level>	Real	$(-5 \times \text{VerticalScale} - \text{OFFSet}) - (5 \times \text{VerticalScale} - \text{OFFSet})$	2V

Remarks

This setting command is only valid when the selected signal source is an analog channel.

Return Format

Query to return the trigger level upper limit in scientific counting form.

Example

```
:TRIGger:SLOPe:ALEVel 0.16      /* Set the upper limit of the trigger level to 160mV */
:TRIGger:SLOPe:ALEVel?          /* Query returns 1.600000 e-01*/
```

3.5.24 :TRIGger:SLOPe:BLEVel

Syntax

```
:TRIGger:SLOPe:BLEVel <level>
```

```
:TRIGger:SLOPe:BLEVel?
```

Description

Set or query the lower limit of the triggering level when the slope is triggered, with the unit consistent with the current amplitude unit.

Parameter

Name	Type	Range	Default
<level>	Real	$(-5 \times \text{VerticalScale} - \text{OFFSet}) - (5 \times \text{VerticalScale} - \text{OFFSet})$	2V

Remarks

This setting command is only valid when the selected signal source is an analog channel.

Return Format

Query to return the lower limit of trigger level in scientific counting form.

Example

```
:TRIGger:SLOPe:BLEVel 0.16 /* Set the trigger level lower limit to 160mV */
```

```
:TRIGger:SLOPe:BLEVel? /* Query returns 1.600000 e-01*/
```

3.5.25 :TRIGger:TIMEout:SOURce

Syntax

```
:TRIGger:TIMEout:SOURce <source>
```

```
:TRIGger:TIMEout:SOURce?
```

Description

Set or query the trigger source triggered by timeout.

Parameter

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

Remarks

When the digital channel is used as the trigger source channel, the digital probe must

be connected to the oscilloscope.

Return Format

Query returns CHAN1 CHAN2.

Example

```
:TRIGger:TIMEout:SOURce CHANnel2      /* Set the trigger source to CH2*/
:TRIGger:TIMEout:SOURce?               /* Query returns CHAN2*/
```

3.5.26 :TRIGger:TIMEout:LEVel

Syntax

```
:TRIGger:TIMEout:LEVel <level>
:TRIGger:TIMEout:LEVel?
```

Description

Set or query the triggering level when timeout is triggered.

Parameter

Name	Type	Range	Default
<level>	Real	$(-5 \times \text{VerticalScale} - \text{OFFSet}) - (5 \times \text{VerticalScale} - \text{OFFSet})$	0

Remarks

This setting command is only valid when the selected signal source is an analog channel.

Return Format

Returns the trigger level value in scientific counting form.

Example

```
:TRIGger:TIMEout:LEVel 0.16           /* Set the trigger level to 160mv */
:TRIGger:TIMEout:LEVel?                /* Query returns 1.600000 e-01*/
```

3.5.27 :TRIGger:TIMEout:WIDth

Syntax

```
:TRIGger:TIMEout:WIDth <value>
:TRIGger:TIMEout:WIDth?
```

Description

Set or query the timeout triggered by timeout. The default unit is s.

Parameter

Name	Type	Range	Default
<value>	Real	8ns-10s	200ns

Remarks

N/A

Return Format

Query returns the timeout value in scientific counting form.

Example

```
:TRIGger:TIMEout:WIDth 0.002      /* Set the timeout time to 2ms */
:TRIGger:TIMEout:WIDth?           /* Query returns 2.000000 e-03*/
```

3.5.28 :TRIGger:TIMEout:POLarity

Syntax

```
:TRIGger:TIMEout:POLarity <polarity>
:TRIGger:TIMEout:POLarity?
```

Description

Set or query the edge type triggered by timeout.

Parameter

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

Remarks

- Positive: Start timing by triggering the level on the rising edge of the input signal.
- NEGative: Start timing by triggering the level on the falling edge of the input signal.

Return Format

Query and return POS or NEG.

Example

```
:TRIGger:TIMEout:POLarity NEGative /* Set the edge type to the falling edge */
:TRIGger:TIMEout:POLarity?         /* Query returns NEG */
```

3.5.29 :TRIGger:WINDOw:SOURce

Syntax

:TRIGger:WINDOw:SOURce <source>

:TRIGger:WINDOw:SOURce?

Description

Set or query the trigger source of the window.

Parameter

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

Remarks

N/A

Return Format

The query returns CHAN1, CHAN2.

Example

:TRIGger:WINDOw:SOURce CHANnel2 /* Set the trigger source to CH2 */

:TRIGger:WINDOw:SOURce? /* Query returns CHAN2 */

3.5.30 :TRIGger:WINDOw:ALEVel

Syntax

:TRIGger:WINDOw:ALEVel <level>

:TRIGger:WINDOw:ALEVel?

Description

Set or query the trigger level when the window is triggered.

Parameter

Name	Type	Range	Default
<level>	full mold	$(-5 \times \text{VerticalScale} - \text{OFFSet}) - (5 \times \text{VerticalScale} - \text{OFFSet})$	0

Remarks

This setting command is only valid when the selected signal source is an analog

channel.

Return Format

Return the trigger level value in scientific notation.

Example

```
:TRIGger:WINDOw:ALEVel 0.16          /* Set the trigger level to 160mv.*/
:TRIGger:WINDOw:ALEVel?              /* Query returns 1.600000e-01*/
```

3.5.31 :TRIGger:WINDOw:BLEVel

Syntax

```
:TRIGger:WINDOw:BLEVel <level>
:TRIGger:WINDOw:BLEVel?
```

Description

Set or query the trigger level when the window is triggered.

Parameter

Name	Type	Range	Default
<level>	full mold	$(-5 \times \text{VerticalScale} - \text{OFFSet}) - (5 \times \text{VerticalScale} - \text{OFFSet})$	0

Remarks

This setting command is only valid when the selected signal source is an analog channel.

Return Format

Return the trigger level value in scientific notation.

Example

```
:TRIGger:WINDOw:BLEVel 0.16          /* Set the trigger level to 160mv.*/
:TRIGger:WINDOw:BLEVel?              /* Query returns 1.600000e-01*/
```

3.5.32 :TRIGger:PATTerN:SOURce

Syntax

```
:TRIGger:PATTerN:SOURce <Source>
:TRIGger:PATTerN:SOURce?
```

Description

Set or query the trigger source for logical triggering.

Parameter

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

Remarks

When the digital channel is used as the trigger source channel, the digital probe must be connected to the oscilloscope.

Return Format

Query returns CHAN1 CHAN2.

Example

```
:TRIGger:PATtern:SOURce CHANnel2    /* Set the trigger signal source to CH2*/
:TRIGger:PATtern:SOURce?             /* Query returns CHAN2*/
```

3.5.33 :TRIGger:PATtern:PATtern

Syntax

```
:TRIGger:PATtern:PATtern <pa_ch1>[,<pa_ch2>]
:TRIGger:PATtern:PATtern?
```

Description

Set or query the code type for each channel when code type triggering is enabled.

Parameter

Name	Type	Range	Default
<pa_ch1>	Discrete	{H L X }	X
<pa_ch2>	Discrete	{H L X }	X

Remarks

- H indicates high level (higher than the threshold level of the channel).
- L indicates low level (lower than the threshold level of the channel).
- X indicates ignore the channel (the channel is not part of the code type; when all channels are set to X, the oscilloscope will not trigger).

Return Format

Query the code type that returns 2 simulated channels or all current channel settings, with multiple channels separated by commas.

Example

```
:TRIGger: PATtern: PATtern H      /* Set the logic type to H */
:TRIGger: PATtern: PATtern?      /* Query returns H */
```

3.5.34 :TRIGger: PATtern: LEVEL

Syntax

```
:TRIGger: PATtern: LEVEL <chan>, <level>
:TRIGger: PATtern: LEVEL? <chan>
```

Description

Set or query the trigger level of the specified channel when setting the code type trigger, with the unit consistent with the current amplitude unit.

Parameter

Name	Type	Range	Default
<level>	full mold	$(-5 \times \text{VerticalScale} - \text{OFFSet}) - (5 \times \text{VerticalScale} - \text{OFFSet})$	0

Remarks

This setting command is only valid when the selected signal source is an analog channel.

Return Format

Return the trigger level value in scientific notation.

Example

```
:TRIGger: PATtern: CODE L      /* Set the code type configuration to L */
:TRIGger: PATtern: CODE?      /* Query returns L */
```

3.5.35 :TRIGger: INTERVal: SOURce

Syntax

```
:TRIGger: INTERVal: SOURce <source>
:TRIGger: INTERVal: SOURce? <chan>
```

Description

Set or query the trigger source of interval triggering.

Parameter

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

Remarks

Only the simulated channel can be used as a trigger source.

Return Format

The query returns CHANnel1, CHANnel2.

Example

```
:TRIGger:INTERVAL:SOURce CHANnel2      /* Set the trigger source to CH2 */
:TRIGger:INTERVAL:SOURce?              /* Query returns CH2 */
```

3.5.36 :TRIGger:INTERVAL:SLOp

Syntax

```
:TRIGger:INTERVAL:SLOp <slop>
:TRIGger:INTERVAL:SLOp?
```

Description

Set or query the edge type of interval-triggered events.

Parameter

Name	Type	Range	Default
<slop>	Discrete	{RISIng FALLIng}	RISIng

Remarks

- RISING: Rising edge trigger.
- FALLING: Falling edge trigger.

Return Format

The query returns RISIng or FALLIng.

Example

```
:TRIGger:INTERVAL:SLOp RISIng      /*Set the edge type to rising edge*/
:TRIGger:INTERVAL:SLOp?            /*Query returns RISIng */
```

3.5.37 :TRIGger:INTERVAL:WHEN

Syntax

```
:TRIGger:INTERVAL:WHEN <when>
:TRIGger:INTERVAL:WHEN?
```

Description

Set or query the trigger conditions for interval triggering.

Parameter

Name	Type	Range	Default
<when>	Discrete	{EQUAL NEQUAL GREATER LESS}	EQUAL

Remarks

- EQUAL = (Equal to time value): Trigger only when the width of the positive pulse or negative pulse of the input signal is equal to the set pulse width. [Pulse width error is 5%]
- NEQUAL != (Not equal to time value): Trigger only when the width of the positive pulse or negative pulse of the input signal is not equal to the set pulse width. [Pulse width error is 5%]
- GREATER > (Greater than time value): Trigger only when the width of the positive pulse or negative pulse of the input signal is greater than the set pulse width. [Pulse width error is 5%]
- LESS < (Less than time value): Trigger only when the width of the positive pulse or negative pulse of the input signal is less than the set pulse width. [Pulse width error is 5%]

Return Format

The query returns EQUAL, NEQUAL, GREATER, LESS.

Example

```
:TRIGger:INTERVAL:WHEN LESS      /*Set the trigger condition to LESS*/
:TRIGger:INTERVAL:WHEN?          /* Query returns LESS */
```

3.5.38 :TRIGger:INTERVAL:TIME

Syntax

```
:TRIGger:INTERVAL:TIME <value>
:TRIGger:INTERVAL:TIME?
```

Description

Set or query the timeout for interval-triggered operations, with the default unit being seconds.

Parameter

Name	Type	Range	Default
<value>	Floating-point	8ns-10s	200ns

Remarks

N/A.

Return Format

Query to return time values in scientific notation.

Example

```
:TRIGger:INTERVAL:TIME 0.002      /*Set the time to 2ms*/
:TRIGger:INTERVAL:TIME?           /* Query returns 2.000000e-03*/
```

3.5.39 **:TRIGger:INTERVAL:ALEVel**

Syntax

```
:TRIGger:INTERVAL:ALEVel <level>
:TRIGger:INTERVAL:ALEVel?
```

Description

Set or query the trigger level when the interval trigger is activated.

Parameter

Name	Type	Range	Default
<level>	Floating-point	$(-5 \times \text{VerticalScale} - \text{OFFSet}) - (5 \times \text{VerticalScale} - \text{OFFSet})$	0

Remarks

This setting command is only valid when the selected information source is a simulated channel.

Return Format

Return the trigger level value in scientific notation.

Example

```
:TRIGger:INTERVAL:ALEVel 0.16      /*Set the trigger level to 160mv*/
:TRIGger:INTERVAL:ALEVel?           /* Query returns 1.600000e-01*/
```

3.5.40 **:TRIGger:UNDER Am:SOURce**

Syntax

```
:TRIGger:UNDER_Am:SOURce <source>
```

:TRIGger:UNDER_Am:SOURce?

Description

Set or query the trigger source of underflow triggering.

Parameter

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

Remarks

Only the simulated channel can be used as a trigger source.

Return Format

The query returns CHANnel1, CHANnel2.

Example

```
:TRIGger:UNDER_Am:SOURce CHANnel2    /* Set the trigger source to CH2*/
:TRIGger:UNDER_Am:SOURce?              /* Query returns CHAN2*/
```

3.5.41 :TRIGger:UNDER Am:POLarity

Syntax

```
:TRIGger:UNDER_Am:POLarity <polarity>
:TRIGger:UNDER_Am:POLarity?
```

Description

Set or query the edge type triggered by underflow.

Parameter

Name	Type	Range	Default
<polarity>	Discrete	{POSitive NEGative}	POSitive

Remarks

- Positive: Positive polarity trigger.
- Negative: Negative polarity trigger.

Return Format

The query returns POSItive, NEGative.

Example

```
:TRIGger:UNDER_Am:POLarity NEGative /*Set the pulse polarity to negative polarity*/
:TRIGger:UNDER_Am:POLarity?          /* Query returns NEG */
```

3.5.42 :TRIGger:UNDER Am:WHEN

Syntax

:TRIGger:UNDER_Am:WHEN <when>

:TRIGger:UNDER_Am:WHEN?

Description

Set or query the trigger conditions for underflow triggers.

Parameter

Name	Type	Range	Default
<when>	Discrete	{EQUAL NEQUAl GREAt LESS }	EQUAl

Remarks

- Equal=(equal to time value): can only be triggered when the positive or negative pulse width of the input signal is equal to the set pulse width. [Pulse width error is 5%]
- NEQUAl= (Not equal to time value): Only when the positive or negative pulse width of the input signal is not equal to the set pulse width, can it trigger 【 Pulse width error is 5% 】 .
- GREAt>(greater than time value): Only when the positive or negative pulse width of the input signal is greater than the set pulse width can it trigger [pulse width error is 5%].
- LESS<(less than time value): Only when the positive or negative pulse width of the input signal is less than the set pulse width can it trigger [pulse width error is 5%].

Return Format

Query returns QUAL, NEQUAl, GREAt, LESS.

Example

:TRIGger:UNDER_Am:WHEN LESS /* Set qualifiers to LESS */

:TRIGger:UNDER_Am:WHEN? /* Query returns LESS */

3.5.43 :TRIGger:UNDER Am:TIME

Syntax

:TRIGger:UNDER_Am:TIME <value>

:TRIGger:UNDER_Am:TIME?

Description

Set or query the timeout for underflow triggers, with the default unit being seconds.

Parameter

Name	Type	Range	Default
<value>	Real	8ns-10s	200ns

Remarks

N/A

Return Format

Query to return the timeout time value in scientific notation.

Example

```
:TRIGger: UNDER_Am:TIME 0.002          /*Set the timeout to 2ms*/
:TRIGger: UNDER_Am:TIME?                /* Query returned 2.000000e-03*/
```

3.5.44 :TRIGger:UNDER Am:ALEVel

Syntax

```
:TRIGger:UNDER_Am:ALEVel <level>
:TRIGger:UNDER_Am:ALEVel?
```

Description

Set or query the trigger level when an underflow trigger occurs.

Parameter

Name	Type	Range	Default
<level>	Real	$(-5 \times \text{VerticalScale} - \text{OFFSet}) - (5 \times \text{VerticalScale} - \text{OFFSet})$	0

Remarks

This setting command is only valid when the selected source is an analog channel.

Return Format

Return the trigger level value in scientific notation.

Example

```
:TRIGger: UNDER_Am:ALEVel 0.16          /*Set the trigger level to 160mv*/
:TRIGger: UNDER_Am:ALEVel?              /* Query returns 1.600000 e-01*/
```

3.5.45 :TRIGger:UNDER Am:BLEVel

Syntax

:TRIGger:UNDER_Am:BLEVel <level>

:TRIGger:UNDER_Am:BLEVel?

Description

Set or query the trigger level when an underflow trigger occurs.

Parameter

Name	Type	Range	Default
<level>	Real	$(-5 \times \text{VerticalScale} - \text{OFFSet}) - (5 \times \text{VerticalScale} - \text{OFFSet})$	0

Remarks

This setting command is only valid when the selected source is an analog channel.

Return Format

Return the trigger level value in scientific notation.

Example

:TRIGger:UNDER_Am:BLEVel 0.16 /*Set the trigger level to 160mv*/

:TRIGger:UNDER_Am:BLEVel? /* Query returns 1.600000 e-01*/

3.5.46 :TRIGger:UART:SOURce

Syntax

:TRIGger:UART:SOURce <source>

:TRIGger:UART:SOURce?

Description

Set or query the trigger source for UART.

Parameter

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

Remarks

N/A.

Return Format

Query returns CHAN1,CHAN2.

Example

```
:TRIGger:UART:SOURce CHANnel1      /* Set the trigger source to CH1*/
:TRIGger:UART:SOURce?              /*Query returns CHAN1*/
```

3.5.47 :TRIGger:UART:CONdition

Syntax

```
:TRIGger:UART:CONdition <condition>
:TRIGger:UART:CONdition?
```

Description

Set or query the conditions for UART trigger.

Parameter

Name	Type	Range	Default
<when>	Discrete	{START STOP READ_DATA PARITY_ERR COM_ERR }	START

Remarks

- START: When the UART start bit appears, it is triggered in the middle of the bit.
- STOP: When the UART stop bit appears, it is triggered in the middle of the bit. Regardless of whether the tested device stops at positions 1, 1.5, or 2, this machine will install one position for processing.
- READDATA: The normal data reception is completed, and the received UART data is equal to the user set data, which is triggered at the stop bit.
- PARITY-ERR: When the data is received normally and there is an error in the parity check of the data, it is triggered at the stop bit.
- COM-ERR: Triggered when an error occurs when the data is not stopped at the beginning of the verification bit.

Return Format

Query returns START, STOP, READ_DATA, PARITY_ERR, COM_ERR.

Example

```
:TRIGger:UART:CONdition START      /* Set the trigger condition to START */
:TRIGger:UART:CONdition?          /* Query returns START */
```

3.5.48 :TRIGger:UART:BAUD

Syntax

:TRIGger:UART:BAUD <baud>

:TRIGger:UART:BAUD?

Description

Set or query the baud rate triggered by UART. The default unit is bps.

Parameter

Name	Type	Range	Default
< baud_rate >	Discrete	{110 300 600 1200 2400 4800 9600 14400 19200 38400 57600 115200 230400 460800 Custom}	Custom

Remarks

N/A

Return Format

Query returns an integer.

Example

:TRIGger:UART:BAUD 4800 /* Set the baud rate to 4.8kbps */

:TRIGger:UART:BAUD? /* Query returns 4800*/

3.5.49 :TRIGger:UART:ALeVel

Syntax

:TRIGger:UART:ALeVel <level>

:TRIGger:UART:ALeVel?

Description

Set or query the triggering level when UART is triggered, with the unit consistent with the current amplitude unit.

Parameter

Name	Type	Range	Default
<level>	Real	(-5 × VerticalScale - OFFSet)- (5 × VerticalScale - OFFSet)	0

Remarks

This setting command is only valid when the selected signal source is an analog channel.

Return Format

Query to return trigger level in scientific counting form.

Example

```
:TRIGger:UART:ALEVel 0.16      /* Set the trigger level to 160mV*/
:TRIGger:UART:ALEVel?          /* Query returns 1.600000 e-01*/
```

3.5.50 **:TRIGger:UART:DATA**

Syntax

```
:TRIGger:UART:DATA <data>
:TRIGger:UART:DATA?
```

Description

Set or query the data value when UART trigger condition is data.

Parameter

Name	Type	Range	Default
<data>	Discrete	0-2 ⁿ -1	0

Remarks

n is the current data width, with a range of 5, 6, 7, or 8.

Return Format

Query returns an integer.

Example

```
:TRIGger:UART:DATA 8      /* Set the data value to 8*/
:TRIGger:UART:DATA?       /* Query returns 8*/
```

3.5.51 **:TRIGger:UART:WIDTH**

Syntax

```
:TRIGger:UART:WIDTH <width>
:TRIGger:UART:WIDTH?
```

Description

Set or query the data bit width when UART trigger condition is set to data.

Parameter

Name	Type	Range	Default
<width>	Discrete	{5 6 7 8}	5

Remarks

N/A

Return Format

Query returns 5, 6, 7, or 8.

Example

```
:TRIGger:UART:WIDTh 5      /* Set the data bit width to 5*/
:TRIGger:UART:WIDTh?      /* Query returns 5*/
```

3.5.52 :TRIGger:UART:PARity

Syntax

```
:TRIGger:UART:PARity <parity>
:TRIGger:UART:PARity?
```

Description

Set or query the verification method when the UART trigger condition is an error frame or verification error.

Parameter

Name	Type	Range	Default
< parity >	Discrete	{NONE ODD EVEN}	NONE

Remarks

When the triggering condition is a verification error, the verification method cannot be set to NONE. In this case, the verification method defaults to odd verification.

Return Format

Query returns NONE ODD or EVEN.

Example

```
:TRIGger:UART:PARity EVEN /* Set the verification method to even verification */
```

```
:TRIGger:UART:PARity?
```

```
/* Query returns EVEN */
```

3.5.53 :TRIGger:LIN:SOURce

Syntax

```
:TRIGger:LIN:SOURce <source>
```

```
:TRIGger:LIN:SOURce?
```

Description

Set or query the trigger source for LIN triggering.

Parameter

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

Remarks

When the digital channel is used as the trigger source channel, the digital probe must be connected to the oscilloscope.

Return Format

Query returns CHAN1,CHAN2.

Example

```
:TRIGger:LIN:SOURce CHANnel1
```

```
/* Set the trigger source to CH1*/
```

```
:TRIGger:LIN:SOURce?
```

```
/* Query returns CHAN1*/
```

3.5.54 :TRIGger:LIN:IDLe

Syntax

```
:TRIGger:LIN:IDLe <idle>
```

```
:TRIGger:LIN:IDLe?
```

Description

Set or query the idle level triggered by LIN.

Parameter

Name	Type	Range	Default
<idle>	Discrete	{LOW HIGH}	HIGH

Remarks

N/A.

Return Format

Query returns LOW, HIGH.

Example

```
:TRIGger:LIN:IDLe LOW          /*Set idle low level*/
:TRIGger:LIN:IDLe?             /* Query returns LOW */
```

3.5.55 :TRIGger:LIN:BAUd

Syntax

```
:TRIGger:LIN:BAUd <baud>
:TRIGger:LIN:BAUd?
```

Description

Set or query the baud rate triggered by LIN. The default unit is bps.

Parameter

Name	Type	Range	Default
<baud>	Discrete	{110 300 600 1200 2400 4800 9600 14400 19200 38400 57600 115200 230400 460800 Custom}	115200

Remarks

N/A

Return Format

Query returns an integer.

Example

```
:TRIGger:LIN:BAUd 4800        /* Set the baud rate to 4.8kbps */
:TRIGger:LIN:BAUd?            /* Query returns 4800 */
```

3.5.56 :TRIGger:LIN:CONdition

Syntax

```
:TRIGger:LIN:CONdition <condition>
:TRIGger:LIN:CONdition?
```

Description

Set or query the triggering conditions for LIN triggering.

Parameter

Name	Type	Range	Default
<condition>	Discrete	<INTERVAL_FIELD SYNC_FIELD ID_FIELD SYNC_CODE_ERROR IDENTIFIER ID_AND_DATA >	INTERVAL_FIELD

Remarks

- The end of the interval field → triggered by the edge after the end of the LIN interval.
- The synchronization field ends → LIN synchronization field data reception is completed and triggered.
- The ID field ends and triggers the completion of data reception in the LINID field.
- Synchronization code error → triggered when LIN synchronization field data reception is completed but the synchronization field data is not equal to 0x55.
- Trigger when the frame ID → LINID field data is received and the ID data is equal to the user's set ID.
- Frame ID and data → LIN data received normally completed, The ID and data are equal to the user settings.

Return Format

Query returns

INTERVAL_FIELD,SYNC_FIELD,ID_FIELD,SYNC_CODE_ERROR,IDENTIFIER,ID_AND_DATA.

Example

```
:TRIGger:LIN:CONdition INTERVAL_FIELD /* Set the trigger condition to end the
interval field */
:TRIGger:LIN:CONdition? /* Query returns INTERVAL-FIELD */
```

3.5.57 :TRIGger:LIN:ID

Syntax

```
:TRIGger:LIN:ID <id>
:TRIGger:LIN:ID?
```

Description

Set or query the identifier triggered by LIN.

Parameter

Name	Type	Range	Default
------	------	-------	---------

Name	Type	Range	Default
< id>	Discrete	1~255	1

Remarks

N/A

Return Format

Query returns an integer.

Example

```
:TRIGger:LIN:ID 25      /* Set identifier to 25*/
:TRIGger:LIN:ID?        /* Query returns 25*/
```

3.5.58 :TRIGger:LIN:DATA

Syntax

```
:TRIGger:LIN:DATA <index> <data>
:TRIGger:LIN:DATA? <index>
```

Description

Set or query the data value of LIN trigger.

Parameter

Name	Type	Range	Default
<data>	Discrete	8-bit	-
<index>	Discrete	Data index 0-3	-

Remarks

N/A

Return Format

Query returns an integer.

Example

```
:TRIGger:LIN:DATA 2,10      /*Set the data at index 2 to 10*/
:TRIGger:LIN:DATA?2        /* Query returns 10*/
```

3.5.59 :TRIGger:LIN:ALEVel

Syntax

```
:TRIGger:LIN:ALEVel <level>
```

```
:TRIGger:LIN:ALEVel?
```

Description

Set or query the triggering level when LIN is triggered.

Parameter

Name	Type	Range	Default
<level>	Real	$(-5 \times \text{VerticalScale} - \text{OFFSet}) - (5 \times \text{VerticalScale} - \text{OFFSet})$	0

Remarks

This setting command is only valid when the channel source of the data cable is an analog channel.

Return Format

Query to return the trigger level value in scientific counting form.

Example

```
:TRIGger:LIN:ALEVel 0.16      /* Set the trigger level to 160mV */
```

```
:TRIGger:LIN:ALEVel?          /* Query returns 1.600000 e-01*/
```

3.5.60 :TRIGger:CAN:SOURce

Syntax

```
:TRIGger:CAN:SOURce <source>
```

```
:TRIGger:CAN:SOURce?
```

Description

Set or query the trigger source triggered by CAN.

Parameter

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

Remarks

N/A.

Return Format

Query returns CHAN1,CHAN2.

Example

```
:TRIGger:CAN:SOURce CHANnel1      /* Set the trigger source to CH1*/
:TRIGger:CAN:SOURce?               /* Query returns CHAN1*/
```

3.5.61 :TRIGger:CAN:IDLe

Syntax

```
:TRIGger:CAN:IDLe <idle>
:TRIGger:CAN:IDLe?
```

Description

Set or query the idle level triggered by CAN.

Parameter

Name	Type	Range	Default
<source>	Discrete	{LOW HIGH}	HIGH

Remarks

N/A.

Return Format

Query returns LOW, HIGH.

Example

```
:TRIGger:CAN:IDLe LOW      /*Set idle low level*/
:TRIGger:CAN:IDLe?         /* Query returns LOW */
```

3.5.62 :TRIGger:CAN:BAUD

Syntax

```
:TRIGger:CAN:BAUD <baud>
:TRIGger:CAN:BAUD?
```

Description

Set or query the baud rate triggered by CAN. The default unit is bps.

Parameter

Name	Type	Range	Default
<baud>	Discrete	<10000 20000 33300 50000 62500 83300 100000 125000 250000 500000 800000 1000000 Custom>	125000

Remarks

N/A

Return Format

Query returns an integer.

Example

```
:TRIGger:CAN:BAUd 4800      /* Set the baud rate to 4.8kbps */
:TRIGger:CAN:BAUd?          /* Query returns 4800*/
```

3.5.63 :TRIGger:CAN:CONdition

Syntax

```
:TRIGger:CAN:CONdition <condition>
:TRIGger:CAN:CONdition?
```

Description

Set or query the triggering conditions for CAN triggering.

Parameter

Name	Type	Range	Default
<condition>	Discrete	<START_BIT REMOTE_FRAME_ID DATA_FRAME_ID FRAME_ID DATAFRAMEID_AND_DATA FRAME_ERROR ALL_ERROR ACK_ERROR OVERLOAD_FRAME >	START_BIT

Remarks

N/A

Return Format

Query returns
 STARP_BIT,REMOTE_FRAME_ID,DATA_FRAME_ID,FRAME_ID,DATAFRAMEID_AND_DATA,FRAME_ERROR,ALL_REEOR ,ACK_ERROR ,OVERLOAD_FRAME.

Example

```
:TRIGger:CAN:CONdition START_BIT      /* Set the trigger condition to START_BIT */
:TRIGger:CAN:CONdition?                /* Query returns STARP_BIT */
```

3.5.64 :TRIGger:CAN:ID

Syntax

:TRIGger:CAN:ID <id>

:TRIGger:CAN:ID?

Description

Set or query the identifier triggered by CAN.

Parameter

Name	Type	Range	Default
<id>	Discrete	0-28	1

Remarks

N/A

Return Format

The query returns an integer.

Example

:TRIGger:CAN:ID 25 /* Set the identifier to 25*/

:TRIGger:CAN:ID? /* Query returns 25*/

3.5.65 :TRIGger:CAN:DATA

Syntax

:TRIGger:CAN:DATA <index><data>

:TRIGger:CAN:DATA? <index>

Description

Set or query the data value of CAN trigger.

Parameter

Name	Type	Range	Default
<data>	Discrete	8 bit	-
<intex>	Discrete	0-3	-

Remarks

N/A

Return Format

The query returns an integer.

Example

```
:TRIGger:CAN:DATA 2,10      /*Set the data with index 2 to 10*/
:TRIGger:CAN:DATA?          /* Query returns 10*/
```

3.5.66 :TRIGger:CAN:ALEVel**Syntax**

```
:TRIGger:CAN:ALEVel <level>
:TRIGger:CAN:ALEVel?
```

Description

Set or query the trigger level when the CAN is triggered.

Parameter

Name	Type	Range	Default
<level>	Real	$(-5 \times \text{VerticalScale} - \text{OFFSet}) - (5 \times \text{VerticalScale} - \text{OFFSet})$	0

Remarks

This setting command is only valid when the channel source of the data line is an analog channel.

Return Format

Query to return the trigger level value in scientific counting form.

Example

```
:TRIGger:CAN:ALEVel 0.16      /* Set the trigger level to 160mV */
:TRIGger:CAN:ALEVel?          /* Query returns 1.600000 e-01*/
```

3.5.67 :TRIGger:SPI:SDA:SOURce**Syntax**

```
:TRIGger:SPI:SDA:SOURce <source>
:TRIGger:SPI:SDA:SOURce?
```

Description

Set or query the channel source for SPI triggered data lines.

Parameter

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

Remarks

The clock source and data source cannot be set to the same signal source at the same time.

Return Format

Query returns CHAN1,CHAN2.

Example

```
:TRIGger:SPI:SDA:SOURce CHANnel2      /* Set the channel source of the data
cable to CH2*/
```

```
:TRIGger:SPI:SDA:SOURce?              /* Query returns CHAN2*/
```

3.5.68 :TRIGger:SPI:SCL:SOURce

Syntax

```
:TRIGger:SPI:SCL:SOURce <source>
```

```
:TRIGger:SPI:SCL:SOURce?
```

Description

Set or query the channel source of the clock triggered by SPI.

Parameter

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

Remarks

The clock source and data source cannot be set to the same signal source at the same time.

Return Format

Query returns CHAN1,CHAN2.

Example

```
:TRIGger:SPI:SCL:SOURce CHANnel1      /* Set the channel source of the clock
line to CH1*/
```

```
:TRIGger:SPI:SCL:SOURce?              /* Query returns CHAN1*/
```

3.5.69 :TRIGger:SPI:SCK

Syntax

```
:TRIGger:SPI:SCK <slope>
```

```
:TRIGger:SPI:SCK?
```

Description

Set or query the Type of clock edge triggered by SPI.

Parameter

Name	Type	Range	Default
<slope>	Discrete	{Rising Falling}	POSitive

Remarks

- Rising: Sample the SDA data at the rising edge of the clock.
- Falling: Sample the SDA data at the falling edge of the clock.

Return Format

Query returns Rising, Falling.

Example

```
:TRIGger:SPI:SCK Falling      /* Set the clock edge type to Falling* /
:TRIGger:SPI:SCK?             /* Query returns Falling */
```

3.5.70 :TRIGger:SPI:WIDTh

Syntax

```
:TRIGger:SPI:WIDTh <width>
```

```
:TRIGger:SPI:WIDTh?
```

Description

Set or query the data bit width of the data channel triggered by SPI.

Parameter

Name	Type	Range	Default
<width>	Integer	4-32	4

Remarks

N/A

Return Format

Query returns an integer.

Example

```
:TRIGger:SPI:WIDTh 10    /*Set the data bit width to 10*/
:TRIGger:SPI:WIDTh?      /* Query returns 10*/
```

3.5.71 :TRIGger:SPI:DATA

Syntax

```
:TRIGger:SPI:DATA <data>
:TRIGger:SPI:DATA?
```

Description

Set or query the data value under SPI trigger.

Parameter

Name	Type	Range	Default
<data>	Discrete	0- ($2^{32}-1$)	-

Remarks

N/A

Return Format

The query returns an integer.

Example

```
:TRIGger:SPI:DATA 20 /* Set the data to 20 */
:TRIGger:SPI:DATA?    /*Query returns 20*/
```

3.5.72 :TRIGger:SPI:MASK

Syntax

```
:TRIGger:SPI:MASK <mask>
:TRIGger:SPI:MASK?
```

Description

Set or query the masking value triggered by SPI.

Parameter

Name	Type	Range	Default
------	------	-------	---------

Name	Type	Range	Default
<data>	Discrete	0- ($2^{32}-1$)	-

Remarks

N/A

Return Format

The query returns an integer.

Example

```
:TRIGger:SPI:MASK 20      /* Set the data to 20 */
:TRIGger:SPI:MASK?         /* Query returns 20 */
```

3.5.73 :TRIGger:SPI:ALEVel

Syntax

```
:TRIGger:SPI:ALEVel <level>
:TRIGger:SPI:ALEVel?
```

Description

Set or query the triggering level of the clock channel when SPI is triggered, with the unit consistent with the current amplitude unit.

Parameter

Name	Type	Range	Default
<level>	Real	$(-5 \times \text{VerticalScale} - \text{OFFSet}) - (5 \times \text{VerticalScale} - \text{OFFSet})$	0

Remarks

This setting command is only valid when the channel source of the clock line is an analog channel.

Return Format

Query to return trigger level in scientific counting form.

Example

```
:TRIGger:SPI:ALEVel 0.16    /* Set the trigger level to 160mV */
:TRIGger:SPI:ALEVel?        /* Query returns 1.600000 e-01 */
```

3.5.74 :TRIGger:SPI:BLEVel

Syntax

:TRIGger:SPI:BLEVel <level>

:TRIGger:SPI:BLEVel?

Description

Set or query the triggering level of the data channel when SPI is triggered, with the unit consistent with the current amplitude unit.

Parameter

Name	Type	Range	Default
<level>	Real	$(-5 \times \text{VerticalScale} - \text{OFFSet}) - (5 \times \text{VerticalScale} - \text{OFFSet})$	0

Remarks

This setting command is only valid when the channel source of the data cable is an analog channel.

Return Format

Query to return trigger level in scientific counting form.

Example

:TRIGger:SPI:BLEVel 0.16 /* Set the trigger level to 160mV */

:TRIGger:SPI:BLEVel? /* Query returns 1.600000 e-01*/

3.5.75 :TRIGger:IIC:SDA:SOURce

Syntax

:TRIGger:IIC:SDA:SOURce <source>

:TRIGger:IIC:SDA:SOURce?

Description

Set or query the channel source of the I2C triggered data line.

Parameter

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

Remarks

The clock source and data source cannot be set to the same signal source at the same time.

Return Format

Query returns CHAN1,CHAN2.

Example

```
:TRIGger:IIC:SDA:SOURce CHANnel1      /* Set the data source to CH1*/
:TRIGger:IIC:SDA:SOURce?                /* Query returns CHAN1*/
```

3.5.76 :TRIGger:IIC:SCL:SOURce

Syntax

```
:TRIGger:IIC:SCL:SOURce <source>
:TRIGger:IIC:SCL:SOURce?
```

Description

Set or query the channel source of the clock triggered by I2C.

Parameter

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

Remarks

The clock source and data source cannot be set to the same signal source at the same time.

Return Format

Query returns CHAN1,CHAN2.

Example

```
:TRIGger:IIC:SCL:SOURce CHANnel2      /* Set the clock source to CH2*/
:TRIGger:IIC:SCL:SOURce?                /* Query returns CHAN2*/
```

3.5.77 :TRIGger:IIC:CONDiton

Syntax

```
:TRIGger:IIC:CONDition <condition>
:TRIGger:IIC:CONDition?
```

Description

Set or query the triggering conditions for IIC triggering.

Parameter

Name	Type	Range	Default
< condition >	Discrete	{START STOP RESTART MISSEDACK ADDRESS A&D}	START

Remarks

- START: Frame start
- STOP: End of frame.
- RESTART: Restart.
- MISSEDACK: Lost Confirmation
- ADDRESS: Finds the set address value and triggers it on the read/write bit.
- A&D: Simultaneously search for the set address value and data value, triggered when both "address" and "data" conditions are met.

Return Format

Query returns START, STOP, RESTART, MISSEDACK, ADDRESS, or A&D.

Example

```
:TRIGger:IIC:CONdition START      /* Set the trigger condition to frame start */
:TRIGger:IIC:CONdition?           /* Query returns START */
```

3.5.78 :TRIGger:IIC:ADDer

Syntax

```
:TRIGger:IIC:ADDer <addr>
:TRIGger:IIC:ADDer?
```

Description

Set or query the address triggered by IIC.

Parameter

Name	Type	Range	Default
<addr>	Integer	0-2 ⁿ -1: 0-127 or 0-1023	0

Remarks

In expression 2ⁿ-1, N is the current address bit width.

Return Format

Query returns an integer.

Example

```
:TRIGger:IIC:ADDer 100      /* Set the address value to 100*/
:TRIGger:IIC:ADDer?         /* Query returns 100*/
```

3.5.79 :TRIGger:IIC:DATA**Syntax**

```
:TRIGger:IIC:DATA <index><data>
:TRIGger:IIC:DATA? <index>
```

Description

Set or query the IIC trigger data value.

Parameter

Name	Type	Range	Default
<data>	Discrete	8 bit	-
<index>	Discrete	0-8	-

Remarks

N/A

Return Format

The query returns an integer.

Example

```
:TRIGger:IIC:DATA 2,10      /*Set the data with index 2 to 10*/
:TRIGger:IIC:DATA?2         /* Query returns 10*/
```

3.5.80 :TRIGger:IIC:ALEVel**Syntax**

```
:TRIGger:IIC:ALEVel <level>
:TRIGger:IIC:ALEVel?
```

Description

Set or query the trigger level of the IIC clock line when triggering, with the unit consistent with the current amplitude unit.

Parameter

Name	Type	Range	Default
------	------	-------	---------

Name	Type	Range	Default
<level>	Real	$(-5 \times \text{VerticalScale} - \text{OFFSet}) - (5 \times \text{VerticalScale} - \text{OFFSet})$	0

Remarks

This setting command is only valid when the channel source of the clock line is an analog channel.

Return Format

Query to return trigger level in scientific counting form.

Example

```
:TRIGger:IIC:ALEVel 0.16      /* Set the trigger level to 160mV */
:TRIGger:IIC:ALEVel?          /* Query returns 1.600000 e-01*/
```

3.5.81 :TRIGger:IIC:BLEVel

Syntax

```
:TRIGger:IIC:BLEVel <level>
:TRIGger:IIC:BLEVel?
```

Description

Set or query the triggering level of the data line when I2C is triggered, with the unit consistent with the current amplitude unit.

Parameter

Name	Type	Range	Default
<level>	Real	$(-5 \times \text{VerticalScale} - \text{OFFSet}) - (5 \times \text{VerticalScale} - \text{OFFSet})$	0

Remarks

For VerticalScale, please refer to: CHANNEL<n>: SCALE command. For OFF Set, please refer to: CHANNEL<n>: OFF Set command. This setting command is only valid when the channel source of the data cable is an analog channel.

Return Format

Query to return trigger level in scientific counting form.

Example

```
:TRIGger:IIC:BLEVel 0.16      /* Set the trigger level to 160mV*/
:TRIGger:IIC:BLEVel?          /* Query returns 1.600000 e-01*/
```

3.6 :Calibration commands

3.6.1 :CALibrate:START

Syntax

:CALibrate:START

Description

The oscilloscope begins to perform self calibration operations.

Parameter

N/A

Remarks

- The self calibration operation can quickly make the oscilloscope reach the optimal working state to obtain the most accurate measurement values.
- Before performing self calibration, please ensure that all channels are not connected to signals until the self calibration operation is completed.
- During the self calibration process, most of the button functions have been disabled.

Return Format

Returns the current calibration status.

Example

:CALibrate:START /* Oscilloscope calibration enabled */

3.6.2 :CALibrate:STATus?

Syntax

:CALibrate:STATus?

Description

Return the current calibration status.

Parameter

N/A

Remarks

N/A

Return Format

N/A

Example

```
:CALibrate:STATus ?      /* Query calibration status */
```

3.6.3 **:CALibrate:QUIT**

Syntax

```
:CALibrate:QUIT
```

Description

Abandon self calibration operation at any time.

Parameter

N/A

Remarks

N/A

Return Format

N/A

Example

```
:CALibrate:QUIT      /* Exit self calibration */
```

3.7 **:MATH Commands**

The MATH command is used to set various calculation functions for waveform between channels.

3.7.1 **:MATH:DISPlay**

Syntax

```
:MATH:DISPlay <bool>
```

```
:MATH:DISPlay?
```

Description

Turn on or off the mathematical operation function or query the status of the mathematical operation function.

Parameter

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

Remarks

N/A

Return Format

Query returns 1 or 0.

Example

```
:MATH:DISPlay ON      /* Open mathematical operation function */
:MATH:DISPlay?        /* Query returns 1*/
```

3.7.2 :MATH:OPERator

Syntax

```
:MATH:OPERator <opt>
:MATH:OPERator?
```

Description

Set or query operators for mathematical operations.

Parameter

Name	Type	Range	Default
<opt>	Discrete	{ADD SUBTract MULTiply DIVision FFT}	ADD

Remarks

When REF is selected as the Parameter for the: MATH: SOURce1 and/or: MATH: SOURce2 commands, this command is used to set the operator for the outer layer operation of the combination operation. The range of<opt>is {ADD|SUBtract|Multiply|DIVision|FFT}.

Return Format

Query returns ADD,SUBT,MULT,DIV,FFT.

Example

```
:MATH:OPERator ADD      /* Set the mathematical operator to sum */
:MATH:OPERator?         /* Query returns ADD */
```

3.7.3 :MATH:SOURce1

Syntax

:MATH:SOURce1 <source>

:MATH:SOURce1?

Description

Set or query the source A for algebraic operations.

Parameter

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

Remarks

For algebraic operations, this command is used to set the source A.

Return Format

Query returns CHAN1,CHAN2.

Example

:MATH:SOURce1 CHANnel2/* Set the source A of algebraic operations to CH1*/

:MATH:SOURce1? /* Query returns CHAN1*/

3.7.4 :MATH:SOURce2

Syntax

:MATH:SOURce2 <source>

:MATH:SOURce2?

Description

Set or query the source B for the outer layer operation of algebraic/combinatorial operations.

Parameter

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

Remarks

This command is only applicable to algebraic operations (with two sources) and

composite operations where the outer operation is an algebraic operation. For composite operations where the outer operation is an algebraic operation, this command is used to set the source B of the outer operation.

Return Format

Query returns CHAN1,CHAN2.

Example

```
:MATH:SOURce2 CHANnel2      /*Set the source B of algebraic operations to CH2*/
:MATH:SOURce2?               /* Query returns CHANnel2*/
```

3.7.5 :MATH:SCALE

Syntax

```
:MATH:SCALE <scale>
:MATH:SCALE?
```

Description

Set or query the vertical range of the operation result, with units related to the currently selected operator and the unit selected by the source.

Parameter

Name	Type	Range	Default
<scale>	Real	The maximum range is 1p to 5T (in steps of 1-2 to 5)	1.00V

Remarks

The adjustable range of the vertical gear is related to the currently selected operator and the gear of the signal channel. For integral and differential operations, it is also related to the current horizontal time base.

Return Format

Retrieve the vertical range of the current calculation result in scientific counting form.

Example

```
:MATH:SCALE 2      /* Set the vertical gear to 2V*/
:MATH:SCALE?       /* Query returns 2.000000 e+00*/
```

3.7.6 :MATH:OFFSet

Syntax

:MATH:OFFSet <offs>

:MATH:OFFSet?

Description

Set or query the vertical offset of the operation result, with units related to the currently selected operator and the unit selected by the source.

Parameter

Name	Type	Range	Default
<offs>	Real	It is related to the vertical gear of the calculation result (-1000 x MathVerticalScale) to (1000 x MathVerticalScale), with a step of MathVerticalScale/50	0.00V

Remarks

MathVerticalScale is the vertical gear of the calculation result, which can be set by the: MATH:SCALE command.

Return Format

Query returns the vertical offset of the current operation result in scientific counting form.

Example

:MATH:OFFSet 2 /* Set the vertical offset to 2V */

:MATH:OFFSet? /* Query returns 2.000000 e+00*/

3.7.7 :MATH:FFT:SOURce

Syntax

:MATH:FFT:SOURce <src>

:MATH:FFT:SOURce?

Description

Set or query the source of FFT operations/filters.

Parameter

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

Remarks

N/A

Return Format

Query returns CHANnel, CHANnel2.

Example

```
:MATH:FFT:SOURce CHANnel1      /* Set the source of FFT operation to CH1*/
:MATH:FFT:SOURce?              /* Query returns Chanel1*/
```

3.7.8 :MATH:FFT:WINDow

Syntax

```
:MATH:FFT:WINDow <window>
:MATH:FFT:WINDow?
```

Description

Set or query window functions for FFT operations.

Parameter

Name	Type	Range	Default
< window >	Discrete	{RECTangle HANNing HAMMING BLACKman TRIangle FLATtop}	RECTangle

Remarks

The use of window functions can effectively reduce spectral leakage effects. Each window function is suitable for measuring different waveforms and needs to be selected based on the measured waveform and its characteristics.

Return Format

Query returns RECT,HANN,HAMM,BLAC,TRI,FLAT.

Example

```
:MATH:FFT:WINDow BLACKman      /* Set the window function for FFT operation to
Blackman */
:MATH:FFT:WINDow?              /* Query returns BLAC */
```

3.7.9 :MATH:FFT:UNIT

Syntax

```
:MATH:FFT:UNIT <unit>
:MATH:FFT:UNIT?
```

Description

Set or query the vertical unit of FFT operation results.

Parameter

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

Remarks

N/A

Return Format

Query returns VRMS or DB.

Example

```
:MATH:FFT:UNIT VRMS      /* Set the vertical unit of FFT operation result to VRMS */
:MATH:FFT:UNIT?          /* Query returns VRMS */
```

3.7.10 :MATH:FFT:HSCale

Syntax

```
:MATH:FFT:HSCale <hsc>
:MATH:FFT:HSCale?
```

Description

Set or query the frequency range of FFT operation results, with a default unit of Hz.

Parameter

Name	Type	Range	Default
<hsc>	Real	{125000 250000 625000 1250000}	625KHz

Remarks

<hsc> can be set to 1/1000, 1/400, 1/200, 1/100, 1/40, or 1/20 of the FFT sampling rate.

When the FFT mode is TRACe, The FFT sampling rate is the screen sampling rate (i.e. 100/horizontal time base).

When FFT mode is MEMory (memory), The FFT sampling rate is the memory sampling rate (ACQuire: SRATe?).

You can observe the detailed information of the spectrum by reducing the frequency range.

Return Format

Query returns the current frequency range in scientific counting form.

Example

```
:MATH:FFT:HSCale 125000      /* Set the frequency range of FFT operation results to
125kHz */
:MATH:FFT:HSCale?           /*Query returned 1.250000e+05*/
```

3.7.11 :MATH:FFT:HCENter

Syntax

```
:MATH:FFT:HCENter <cent>
:MATH:FFT:HCENter?
```

Description

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

Parameter

Name	Type	Range	Default
<cent>	Real	Reference Marks	125MHz

Remarks

- When the FFT mode is TRACe (trajectory), the range of<center>is 0 to (0.4 x FFT sampling rate); Among them, The FFT sampling rate is the screen sampling rate (i.e. 100/horizontal time base). When the FFT mode is MEMory (memory), the range of<center>is 0 to (0.5 x FFT sampling rate); Among them, The FFT sampling rate is the memory sampling rate (ACQuire: SRAtE?).
- The set step is the horizontal gear/50 of the current FFT calculation result.

Return Format

Query returns the current center frequency in scientific counting form.

Example

```
:MATH:FFT:HCENter 10000000    /* Set the center frequency of the FFT operation
result to 10MHz */
:MATH:FFT:HCENter?           /* Query returns 1.000000 e+07*/
```

3.8 Waveform Data Command Subsystem

3.8.1 :WAVEform:DATA:ALL? <source>

Syntax

:WAVEform:DATA:ALL? <source>

Description

Read waveform data.

Parameter

Name	Type	Range	Default
<source>	Discrete	{Channel1 Channel2 }	Channel1

Remarks

Set the measurement source for reading the waveform, which can be Channel 1-2.

data[0]-data[1] (2 bytes): #9 data[2]-data[10] (9 bytes): Represents the byte length of the current data packet. data[11]-data[19] (9 bytes): Represents the total byte length of the data volume. data[20]-data[28] (9 bytes): Represents the byte length of the uploaded data. data[29] (1 byte): Operation status, where 1 represents running and 0 represents paused, indicated as a string. data[30] (1 byte): Represents the trigger status. data[31-34] (4 bytes): Represents the offset of Channel 1, indicated as a 32-bit integer, unit: uv; in binary form, with the lower byte first. data[35-38] (4 bytes): Represents the offset of Channel 2, indicated as a 32-bit integer, unit: uv; in binary form, with the lower byte first. data[39-42] (4 bytes): Represents the offset of Channel 3, indicated as a 32-bit integer, unit: uv; in binary form, with the lower byte first. data[43-46] (4 bytes): Represents the offset of Channel 4, indicated as a 32-bit integer, unit: uv; in binary form, with the lower byte first.

data[47]-data[53] (7 bytes): Represents the vertical range of Channel 1; string format. If it is a voltage range, the unit is V; if it is a current range, the unit is A.

data[54]-data[60] (7 bytes): Represents the vertical range of Channel 2; string format. If it is a voltage range, the unit is V; if it is a current range, the unit is A.

data[61]-data[67] (7 bytes): Represents the vertical range of Channel 3; string format. If it is a voltage range, the unit is V; if it is a current range, the unit is A.

data[68]-data[74] (7 bytes): Represents the vertical range of Channel 4; string format. If it is a voltage range, the unit is V; if it is a current range, the unit is A.

data[75] (1 byte): Represents channel enable status.

0 indicates Channels 1 and 2 are disabled;

1 indicates only Channel 2 is enabled;

2 indicates only channel 1 is enabled; 3 indicates channels 1 and 2 are enabled;

represented as a string. data[76] (1 byte): indicates channel enable status. 0 indicates channels 3 and 4 are disabled; 1 indicates only channel 4 is enabled; 2 indicates only channel 3 is enabled; 3 indicates channels 3 and 4 are enabled; represented as a string.
 data[77] (1 byte): indicates channel enable status. 0 indicates LA1 and LA2 channels are disabled; 1 indicates only LA2 channel is enabled; 2 indicates only LA1 channel is enabled; 3 indicates LA1 and LA2 channels are enabled; represented as a string.
 data[78] (1 byte): indicates channel enable status. 0 indicates LA3 and LA4 channels are disabled; 1 indicates only LA4 channel is enabled; 2 indicates only LA3 channel is enabled; 3 indicates LA3 and LA4 channels are enabled; represented as a string.
 data[79]-data[87] (9 bytes): Represents the sampling rate, string format.
 data[88]-data[93] (6 bytes): Represents the sampling multiple, which is always 1x, string format.
 data[94]-data[101] (8 bytes): Equivalent to horizontal offset, unit ps, binary format.
 data[102]: Reserved.
 data[103]-data[110] (8 bytes): Pre-trigger time length, unit ps, binary format.
 data[111]-data[127]: Reserved.
 data[128]-data[x]: Represents the waveform data corresponding to the current data header, binary format.

The voltage range in the header are all original ranges without multiplying the probe ratio. When there is a probe ratio, the actual voltage value is calculated by multiplying the voltage value in the header by the corresponding channel's probe ratio: Actual voltage value = Header voltage value * corresponding channel probe ratio / 24 * (data value - 128) - offset value * probe ratio. Each byte of oscilloscope data represents one point data; each byte of logic analyzer data represents LA<n> channel data; the high 4 bits are invalid;

Return Format

N/A.

Example

```
:WAVEform:DATA:ALL?      /*Read waveform data*/
```

3.9 :DISPlay Commands

The DISPlay command can set the waveform display type, afterglow time, waveform brightness, screen display grid type, and grid brightness.

3.9.1 :DISPlay:TYPE

Syntax

```
:DISPlay:TYPE <type>
:DISPlay:TYPE?
```

Description

Set or query the display method of waveforms on the screen.

Parameter

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

Remarks

- VECTors: Displayed by connecting lines between sampling points. This mode provides the most realistic waveform in most cases. It is convenient to view the steep edges of waveforms (such as square waves).
- DOTS: Directly display sampling points. You can visually see each sampling point and use the cursor to measure the X and Y values of that point.

Return Format

Query returns 0 or 1.

Example

```
:DISPlay:TYPE 1      /* Select point display method */
:DISPlay:TYPE?       /* Query returns 1*/
```

3.9.2 :DISPlay:WBRightness

Syntax

```
:DISPlay:WBRightness <time>
:DISPlay:WBRightness?
```

Description

Set or query the brightness of waveform display on the screen.

Parameter

Name	Type	Range	Default
<time>	Integer	0-100	100

Remarks

N/A

Return Format

Query returns an integer between 0 and 100.

Example

```
:DISPlay:WBRightness 50      /* Set waveform brightness to 50%*/
:DISPlay:WBRightness?        /* Query returns 50*/
```

3.9.3 :DISPlay:GRID

Syntax

```
:DISPlay:GRID <grid>
:DISPlay:GRID?
```

Description

Set or query the grid type displayed on the screen.

Parameter

Name	Type	Range	Default
<grid>	Discrete	{{0 DOT} {1 LINE} {2 CLOSE }}	1

Remarks

- DOT: Grid Type is displayed as points.
- LINE: The grid type is displayed in lines.
- Close: Close the background grid.

Return Format

Query returns 0, 1 and 2.

Example

```
:DISPlay:GRID 2      /* Close background grid and coordinates */
:DISPlay:GRID?        /*Query returns 2*/
```

3.9.4 :DISPlay:GBRightness

Syntax

```
:DISPlay:GBRightness <brightness>
:DISPlay:GBRightness?
```

Description

Set or query the brightness of the screen grid.

Parameter

Name	Type	Range	Default
<brightness>	Integer	0-100	0

Remarks

N/A

Return Format

Query returns an integer between 0 and 100.

Example

```
:DISPlay:GBrightness 60      /* Set the screen grid brightness to 60%*/
:DISPlay:GBrightness?        /* Query returns 60*/
```

3.10 :CURSor Commands

The CURSOR command is used to measure the X-axis value (such as time) and Y-axis value (such as voltage) of the screen waveform.

3.10.1 :CURSor:MODE

Syntax

```
:CURSor:MODE <mode>
:CURSor:MODE?
```

Description

Set or query the mode of cursor measurement.

Parameter

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

Remarks

- OFF: Turn off the cursor measurement function.
- MANual: Turn on the manual cursor measurement mode.
- TRACk: Turn on the cursor tracking measurement mode.

Return Format

Query returns OFF, MAN, TRAC.

Example

```
:CURSor:MODE MANual      /* Select manual cursor measurement mode */
:CURSor:MODE?            /* Query returns MAN */
```

3.10.2 :CURSor:MANual:TYPE

Syntax

:CURSor:MANual:TYPE <type>

:CURSor:MANual:TYPE?

Description

Set or query the cursor type for manual cursor measurement.

Parameter

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

Remarks

- X: Select the X-shaped cursor. The X-shaped cursor consists of a vertical solid line (cursor A) and a vertical dashed line (cursor B), commonly used to measure time parameters.
- Y: Select the Y-shaped cursor. The Y-shaped cursor consists of a horizontal solid line (cursor A) and a horizontal dashed line (cursor B), commonly used to measure voltage parameters.

Return Format

Query returns X or Y.

Example

:CURSor:MANual:TYPE X /* Select X-shaped cursor */

:CURSor:MANual:TYPE? /* Query returns X */

3.10.3 :CURSor:MANual:SOURce

Syntax

:CURSor:MANual:SOURce <source>

:CURSor:MANual:SOURce?

Description

Set or query the channel source for manual cursor measurement.

Parameter

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

Remarks

Only the currently open channel can be selected as the channel source.

Return Format

Query returns CHAN1, CHAN2.

Example

```
:CURSor:MANual:SOURce CHANnel2      /* Set channel source to CH2*/
:CURSor:MANual:SOURce?                /* Query returns CHAN2*/
```

3.10.4 **:CURSor:MANual:AX**

Syntax

```
:CURSor:MANual:AX <x>
:CURSor:MANual:AX?
```

Description

Set or query the horizontal position of cursor A during manual cursor measurement.

Parameter

Name	Type	Range	Default
<x>	Integer	0-770	600

Remarks

The cursor's horizontal and vertical positions are defined by the pixel coordinates of the screen.

Return Format

Query returns an integer between 0 and 770.

Example

```
:CURSor:MANual:AX 200      /* Set the horizontal position of cursor A to 200*/
:CURSor:MANual:AX?          /* Query returns 200*/
```

3.10.5 **:CURSor:MANual:AXValue?**

Syntax

```
:CURSor:MANual:AXValue?
```

Description

When querying manual cursor measurements, the X value at cursor A. The unit is determined by the currently selected horizontal unit.

Parameter

N/A

Remarks

N/A

Return Format

Query returns the X value at the current cursor A in scientific counting form.

Example

```
:CURSor:MANual:AXValue?          /* Query returned -4.000000 e-06*/
```

3.10.6 :CURSor:MANual:BX

Syntax

```
:CURSor:MANual:BX <x>
```

```
:CURSor:MANual:BX?
```

Description

Set or query the horizontal position of cursor B during manual cursor measurement.

Parameter

Name	Type	Range	Default
<x>	Integer	0-770	400

Remarks

The cursor's horizontal and vertical positions are defined by the pixel coordinates of the screen.

Return Format

Query returns an integer between 0 and 770.

Example

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

```
:CURSor:MANual:BX?            /* Query returns 200*/
```

3.10.7 :CURSor:MANual:BXValue?

Syntax

:CURSor:MANual:BXValue?

Description

When querying manual cursor measurements, the X value at cursor B. The unit is determined by the currently selected horizontal unit.

Parameter

N/A

Remarks

N/A

Return Format

Query returns the X value at the current cursor B in scientific counting form.

Example

:CURSor:MANual:BXValue? /* Query returns 4.000000e-06*/

3.10.8 :CURSor:MANual:AY

Syntax

:CURSor:MANual:AY <y>

:CURSor:MANual:AY?

Description

Set or query the vertical position of cursor A during manual cursor measurement.

Parameter

Name	Type	Range	Default
<y>	Integer	0-400	179

Remarks

The cursor's horizontal and vertical positions are defined by the pixel coordinates of the screen.

Return Format

The query returns an integer between 0 and 400.

Example

```
:CURSor:MANual:AY 200      /* Set the vertical position of cursor A to 200*/
:CURSor:MANual:AY?         /* Query returns 200*/
```

3.10.9 :CURSor:MANual:AYValue?**Syntax**

```
:CURSor:MANual:AYValue?
```

Description

When querying manual cursor measurements, the Y value at cursor A. The unit is determined by the currently selected vertical unit.

Parameter

N/A

Remarks

N/A

Return Format

When selecting source CHANnel1 or CHANnel2, query the Y value at the current cursor position A in scientific notation.

Example

```
:CURSor:MANual:AYValue?      /* Query returns 2.000000 e+00*/
```

3.10.10 :CURSor:MANual:BY**Syntax**

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

Description

When setting or querying manual cursor measurements, the vertical position of cursor B.

Parameter

Name	Type	Range	Default
<y>	Integer	0-400	299

Remarks

The cursor's horizontal and vertical positions are defined by the pixel coordinates of the screen.

Return Format

The query returns an integer between 0 and 400.

Example

```
:CURSor:MANual:BY 200      /* Set the vertical position of cursor B to 200*/
:CURSor:MANual:BY?         /* Query returns 200*/
```

3.10.11 **:CURSor:MANual:BYValue?**

Syntax

```
:CURSor:MANual:BYValue?
```

Description

When querying manual cursor measurements, the Y value at cursor B. The unit is determined by the currently selected vertical unit.

Parameter

N/A

Remarks

N/A

Return Format

When selecting source CHANnel1 or CHANnel2, query the Y value at the current cursor position B, which is returned in scientific notation.

Example

```
:CURSor:MANual:BYValue?      /* Query returned -2.000000 e+00*/
```

3.10.12 **:CURSor:TRACk:SOURceA**

Syntax

```
:CURSor:TRACk:SOURceA <source>
:CURSor:TRACk:SOURceA?
```

Description

When setting up or querying cursor tracking measurements, the channel source for Cursor A measurement.

Parameter

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

Remarks

Only the currently open channel can be selected as the channel source.

Return Format

The query returns CHAN1, CHAN2.

Example

```
:CURSor:TRACk:SOURceA CHANnel1      /* Set channel source CH1*/
:CURSor:TRACk:SOURceA?                /* Query returns CH1*/
```

3.10.13 :CURSor:TRACk:SOURceB

Syntax

```
:CURSor:TRACk:SOURceB <source>
:CURSor:TRACk:SOURceB?
```

Description

When setting up or querying cursor tracking measurements, the channel source for Cursor B measurement.

Parameter

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

Remarks

Only the currently open channel can be selected as the channel source.

Return Format

The query returns CHAN1, CHAN2.

Example

```
:CURSor:TRACk:SOURceB CHANnel1      /* Set channel source CH1*/
:CURSor:TRACk:SOURceB?                /* Query returns CH1*/
```

3.10.14 :CURSor:TRACk:AX

Syntax

:CURSor:TRACk:AX <x>

:CURSor:TRACk:AX?

Description

Set or query the horizontal position of cursor A when tracking measurements.

Parameter

Name	Type	Range	Default
<x>	Integer	0-770	400

Remarks

The cursor's horizontal and vertical positions are defined by the pixel coordinates of the screen.

Return Format

Query returns an integer between 0 and 770.

Example

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

:CURSor:TRACk:AX? /* Query returns 200*/

3.10.15 :CURSor:TRACk:AXValue?

Syntax

:CURSor:TRACk:AXValue?

Description

When querying cursor tracking measurement, get the X value at Cursor A. The default unit is s.

Parameter

N/A

Remarks

N/A

Return Format

Query to return the X value at cursor A in scientific notation.

Example

```
:CURSor:TRACK:AXValue?          /* Query returned -4.000000e-06*/
```

3.10.16 :CURSor:TRACK:AY

Syntax

```
:CURSor:TRACK:AY <y>
```

```
:CURSor:TRACK:AY?
```

Description

When querying cursor tracking measurement, get the vertical position of cursor A.

Parameter

Name	Type	Range	Default
<y>	Integer	0-400	179

Remarks

The cursor's horizontal and vertical positions are defined by the pixel coordinates of the screen.

Return Format

Query returns an integer between 0 and 400.

Example

```
:CURSor:TRACK:AY 200          /*Set the vertical position of cursor A to 200 */
```

```
:CURSor:TRACK:AY?            /* Query returns 200*/
```

3.10.17 :CURSor:TRACK:AYValue?

Syntax

```
:CURSor:TRACK:AYValue?
```

Description

When querying manual cursor measurement, get the Y value at cursor A. The unit is determined by the currently selected vertical unit.

Parameter

N/A

Remarks

N/A

Return Format

When selecting source CHANnel1 or CHANnel2, query the Y value at the current cursor position A, which is returned in scientific notation.

Example

```
:CURSor:TRACK:AYValue?          /* Query returned 2.000000e+00*/
```

3.10.18 :CURSor:TRACK:BX

Syntax

```
:CURSor:TRACK:BX <x>
```

```
:CURSor:TRACK:BX?
```

Description

Set or query the horizontal position of cursor B when tracking cursor measurements.

Parameter

Name	Type	Range	Default
<x>	Integer	0-770	600

Remarks

The cursor's horizontal and vertical positions are defined by the pixel coordinates of the screen.

Return Format

The query returns an integer between 0 and 770.

Example

```
:CURSor:TRACK:BX 200          /* Set the horizontal position of cursor B to 200*/
```

```
:CURSor:TRACK:BX?            /* Query returns 200*/
```

3.10.19 :CURSor:TRACK:BXValue?

Syntax

```
:CURSor:TRACK:BXValue?
```

Description

When querying cursor tracking measurement, get the X value at Cursor B. The default unit is s.

Parameter

N/A

Remarks

N/A

Return Format

Query to return the X value at cursor B in scientific notation.

Example

```
:CURSor:TRACK:BXValue?          /* Query returned 4.000000e-06*/
```

3.10.20 :CURSor:TRACk:BY

Syntax

```
:CURSor:TRACk:BY <y>
```

```
:CURSor:TRACk:BY?
```

Description

When querying cursor tracking measurement, get the vertical position of cursor B.

Parameter

Name	Type	Range	Default
<y>	Integer	0-400	299

Remarks

The cursor's horizontal and vertical positions are defined by the pixel coordinates of the screen.

Return Format

Query returns an integer between 0 and 400.

Example

```
:CURSor:TRACk:BY 200          /*Set the vertical position of cursor B to 200 */
:CURSor:TRACk:BY?              /* Query returns 200*/
```

3.10.21 :CURSor:TRACk:BYValue?

Syntax

:CURSor:TRACk:BYValue?

Description

When querying manual cursor measurement, get the Y value at cursor B. The unit is determined by the currently selected vertical unit.

Parameter

N/A

Remarks

N/A

Return Format

When selecting source CHANnel1 or CHANnel2, query the Y value at the current cursor position B, which is returned in scientific notation.

Example

:CURSor:TRACk:BYValue? /* Query returned 2.000000e+00 */

3.11 :MEASure Commands

3.11.1 :MEASure:ENABLE

Syntax

:MEASure:ENABLE <bool>

:MEASure:ENABLE?

Description

Set or query the source of information for the current measurement parameter.

Parameter

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

Remarks

N/A

Return Format

The query returns ON or OFF.

Example

```
:MEASure:ENABLE ON      /*Open measurement*/
:MEASure:ENABLE?         /* Query returns ON*/
```

3.11.2 :MEASure:SOURce

Syntax

```
:MEASure:SOURce <source>
:MEASure:SOURce?
```

Description

Set or query the source of information for the current measurement parameter.

Parameter

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

Remarks

Only the currently open channels are optional.

Return Format

Query and return CHAN1, CHAN2.

Example

```
:MEASure:SOURce CHANnel2 /* Set the Parameter measurement source to CH2*/
:MEASure:SOURce?         /* Query returns CHAN2*/
```

3.11.3 :MEASure:ADISplay

Syntax

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

Description

Turn all measurements on or off, or query the current status of all measurements.

Parameter

Name	Type	Range	Default
<sour>	Discrete	{{1 ON}}{{0 OFF}}	0

Remarks

N/A.

Return Format

Query and return ON, OFF.

Example

```
:MEASure:ADISplay ON          /*Open all measurements*/
:MEASure:ADISplay?            /* Query returns ON*/
```

3.11.4 :MEASure:ITEM

Syntax

```
:MEASure:CHANnel<n>:ITEM <type>
:MEASure:CHANnel<n>:ITEM?
```

Description

Query the measurement results of the specified parameters.

Parameter

Name	Type	Range	Default
<item>	Discrete	{ FREQuency PERiod VAVG VMAX VMIN VPP VTOP VMID VBASE VAMP RMS OVERshoot P REShoot PERIodRMS PERIodMEAN RTIME FTI Me PPULess NPULses PDUTy NDUTy FRR FFF FOVshoot RPRESshoot BNIDth FRF FFR LRR L RF LFR LFF	---

Remarks

N/A.

Return Format

Query returns the current measurement value in scientific counting form.

Example

```
:MEASure:CHANnel1:ITEM VPP /*Open the overshoot measurement for channel1*/
:MEASure:CHANnel1:ITEM? VPP /* Query returned 8.888889e-03*/
```

3.11.5 :MEASure:GATE:ENABle

Syntax

:MEASure:GATE:ENABle <bool>

:MEASure:GATE:ENABle?

Description

Set or query the current door lock status.

Parameter

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

Remarks

N/A.

Return Format

The query returns ON or OFF.

Example

:MEASure:GATE:ENABle ON /*Open measurement */

:MEASure:GATE:ENABle? /* Query returned ON*/

3.11.6 :MEASure:GATE:AY

Syntax

:MEASure:GATE:AY <value>

:MEASure:GATE:AY?

Description

Set or query the value of cursor A (horizontal position).

Parameter

Name	Type	Range	Default
<value>	real	0-400	179

Remarks

N/A.

Return Format

The query returns an integer between 0 and 400.

Example

```
:MEASure:GATE:AY 200    /* Set the value of cursor A to 200*/
:MEASure:GATE:AY?       /* Query returned 200*/
```

3.11.7 :MEASure:GATE:BY**Syntax**

```
:MEASure:GATE:BY <value>
:MEASure:GATE:BY?
```

Description

Set or query the value of cursor B (horizontal position).

Parameter

Name	Type	Range	Default
<value>	real	0-400	179

Remarks

N/A.

Return Format

The query returns an integer between 0 and 400.

Example

```
:MEASure:GATE:BY 200    /* Set the value of cursor B to 200*/
:MEASure:GATE:BY?       /* Query returned 200*/
```

3.12 :MASK Commands

The MASK command is used to set and query the relevant parameters in pass/fail tests.

3.12.1 :MASK:ENABle**Syntax**

```
:MASK:ENABle <bool>
:MASK:ENABle?
```

Description

Turn on or off the pass/fail testing function, or query the status of the pass/fail testing function.

Parameter

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

Remarks

The pass/fail test function is invalid in the following situations: horizontal time base mode in XY or Roll mode, slow scan mode (i.e. horizontal time base in YT mode, horizontal time base in 200ms/div or slower), and waveform recording.

Return Format

Query returns 1 or 0.

Example

```
:MASK:ENABLE ON      /* Enable pass/fail testing function */
:MASK:ENABLE?         /* Query returns 0*/
```

3.12.2 :MASK:SOURce

Syntax

```
:MASK:SOURce <source>
:MASK:SOURce?
```

Description

Set or query the measurement source for pass/fail tests.

Parameter

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

Remarks

This command can only set open channels and can send the: CHANnel<n>: DISPlay command to open the desired channel.

Return Format

Query returns CHAN1,CHAN2.

Example

```
:MASK:SOURce CHANnel2      /* Set the measurement source for pass/fail tests to
CH2*/
```

:MASK:SOURce? /* Query returns CHAN2*/

3.12.3 :MASK:MDISplay

Syntax

:MASK:MDISplay <bool>

:MASK:MDISplay?

Description

When the pass/fail test is opened, turn on or off statistical information, or query the status of statistical information.

Parameter

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

Remarks

- Before executing this command, it is necessary to send the MASK: ENABLE command to enable the pass/fail test function.
- When the statistical information is opened, the test results shown in the following figure will be displayed in the upper right corner of the screen.
- Can send: MASK: PASSed?, : MASK: FAILED? And: MASK: TOTAL? Command query test results.

Return Format

Query returns ON or OFF.

Example

:MASK:MDISplay ON /* Open Statistics */

:MASK:MDISplay? /* Query returns 1*/

3.12.4 :MASK:OUTPut

Syntax

:MASK:OUTPut <bool>

:MASK:OUTPut?

Description

Turn on or off the sound prompt when testing fails, or check the status of the sound prompt.

Parameter

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

Remarks

- Off: When a failed waveform is detected, there is a display and output, but the buzzer does not sound an alarm, and the buzzer is turned off.
- Open: When a failed waveform is detected, there is a display and output, and the buzzer emits an audible alarm (independent of the sound switch status), and the buzzer opens.

Return Format

Query returns 1 and 0.

Example

```
:MASK:OUTPut ON      /* Turn on sound prompt (buzzer on)*/
:MASK:OUTPut?        /* Query returns 1*/
```

3.12.5 :**MASK:SOOutput**

Syntax

```
:MASK:SOOutput <bool>
:MASK:SOOutput?
```

Description

Turn on or off stop output, or query the status of stop output.

Parameter

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

Remarks

- Open: When a failed waveform is detected, the oscilloscope will stop testing and enter the "STOP" state. At this point, the screen remains displaying the measurement results (if the display is turned on), and the rear panel [Trigger Out] (if enabled) only outputs one pulse.
- Off: Even if a failed waveform is detected, the oscilloscope will continue to test, and the test results on the screen are constantly updated. The Trigger Out on the rear panel will output pulses every time a failed waveform is detected.

Return Format

Query returns ON and OFF.

Example

```
:MASK:SOOutput ON      /* Turn on stop output */
:MASK:SOOutput?        /* Query returns 1*/
```

3.12.6 :MASK:X

Syntax

```
:MASK:X <value>
:MASK:X?
```

Description

Set or query the horizontal adjustment parameters in the pass/fail test rules, with a default unit of div.

Parameter

Name	Type	Range	Default
<x>	Real	0.02 to 4, with a step of 0.01 within the range of values	0.24

Remarks

N/A

Return Format

Query returns the current horizontal adjustment parameter in scientific counting form.

Example

```
:MASK:X 0.28      /* Set the horizontal adjustment parameter to 0.28div */
:MASK:X?          /* Query returned 2.800000 e-01*/
```

3.12.7 :MASK:Y

Syntax

```
:MASK:Y <value>
:MASK:Y?
```

Description

Set or query the vertical adjustment parameters in the pass/fail test rules, with a default unit of div.

Parameter

Name	Type	Range	Default
<y>	Real	0.04 to 5.12, with a step of 0.01 within the range of values	0.04

Remarks

N/A

Return Format

Query returns the current vertical adjustment parameter in scientific counting form.

Example

```
:MASK:Y 0.36          /* Set the vertical adjustment parameter to 0.36div */
:MASK:Y?              /* Query returned 3.600000 e-01*/
```

3.12.8 :MASK:CREate

Syntax

:MASK:CREate

Description

Create a pass/fail test rule by adjusting the parameters horizontally and vertically based on the current settings.

Parameter

N/A

Remarks

This command is only valid when the pass/fail test function is turned on (: MASK: ENABLE) and not running (: MASK: OPERate).

Return Format

N/A

Example

N/A

3.13 :SYSTem Commands

3.13.1 :SYSTem:GAM?

Syntax

:SYSTem:GAM?

Description

Query the number of grids in the horizontal direction of the instrument screen.

Parameter

N/A

Remarks

The query returns a fixed 12.

Return Format

N/A

Example

N/A

3.13.2 :SYSTem:RAM?

Syntax

:SYSTem:RAM?

Description

Query the number of analog channels of the instrument.

Parameter

N/A

Remarks

The query returns a fixed 2.

Return Format

N/A

Example

N/A

3.13.3 :SYSTem:PON**Syntax**

:SYSTem:PON <value>

:SYSTem:PON?

Description

Set or query the configuration type called when the oscilloscope is powered on again.

Parameter

Name	Type	Range	Default
<value>	Real	<LATEST DEFAULT>	DEFAULT

Remarks

N/A

Return Format

The query returns LATEST, DEFAULT.

Example

```
:SYSTem:PON LATEST          /* Set the oscilloscope to restore the last values when
re-powered */
```

```
:SYSTem:PON?                /* Query returned LATEST */
```

3.13.4 :SYSTem:LANGuage**Syntax**

:SYSTem:LANGuage <value>

:SYSTem:LANGuage?

Description

Set or query the system display language.

Parameter

Name	Type	Range	Default
<value>	Real	<ENGLISH SCHINESE>	ENGLISH

Remarks

N/A

Return Format

The query returns ENGLISH, SCHinese.

Example

```
:SYSTem:LANGUage SCHinese      /* Set the system language to Chinese */
:SYSTem:LANGUage?              /* Query returned SCHinese */
```

3.13.5 **:SYSTem:LOCKed**

Syntax

```
:SYSTem:LOCKed <bool>
:SYSTem:LOCKed?
```

Description

Turn on or off the keyboard lock function, or check the status of the keyboard lock function.

Parameter

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

Remarks

N/A

Return Format

The query returns ON, OFF.

Example

```
:SYSTem:LOCKed ON      /* Turn on keyboard lock function*/
:SYSTem:LOCKed?        /* Query returned ON */
```

3.14 **[:SOURce] Commands**

The [:SOURce[<n>]] command is used to set parameters related to the built-in signal source. When <n> is 1, it refers to the corresponding built-in signal source channel. If <n> is omitted or :SOURce[<n>] is used, the default operation is performed on signal source 1. This setting is only supported on machines with signal source functionality.

3.14.1 :DDS:SWITCh

Syntax

:DDS:SWITCh <bool>

:DDS:SWITCh?

Description

Set or query the signal source status.

Parameter

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

Remarks

N/A

Return Format

Query returns OFF or ON.

Example

:DDS:SWITCh ON /*Turn on keyboard lock function* /

:DDS:SWITCh? /* Query returns ON */

3.14.2 :DDS:TYPE

Syntax

:DDS:TYPE <type>

:DDS:TYPE?

Description

Set or query the waveform type of the signal source.

Parameter

Name	Type	Range	Default
<type>	Real	{SINE SQUAre RAMP EXP NOISe DC ARB1 ARB2 ARB3 ARB4}	SINE

Remarks

N/A

Return Format

Query returns SINE,SQUAre,RAMP,EXP,NOISe,DC,ARB1,ARB2,ARB3,ARB4.

Example

```
:DDS:TYPE SINE      /*Set the signal source waveform type to sine wave */
:DDS:TYPE?          /* Query returns SINE*/
```

3.14.3 :DDS:FREQ

Syntax

```
:DDS:FREQ <freq>
:DDS:FREQ?
```

Description

Set or query the frequency of the signal source signal.

Parameter

Name	Type	Range	Default
<freq>	Real	0Hz-25MHz	1kHz

Remarks

N/A

Return Format

Query to return frequency in scientific notation.

Example

```
:DDS:FREQ 1000      /* Set the output frequency of source 1 to 1kHz */
:DDS:FREQ?          /* Query returned 1.0000000 e+03*/
```

3.14.4 :DDS:AMP

Syntax

```
:DDS:AMP <amp>
:DDS:AMP?
```

Description

Set or query the amplitude of the signal source signal.

Parameter

Name	Type	Range	Default
------	------	-------	---------

Name	Type	Range	Default
<amp>	Real	0-7V	1.5V

Remarks

N/A

Return Format

Query to return the amplitude in scientific notation.

Example

```
:DDS:AMP 1          /*Set the signal source waveform amplitude to 1V */
:DDS:AMP?           /* Query returned 1.000000 e+00*/
```

3.14.5 :DDS:OFFSet

Syntax

```
:DDS:OFFSet <offset>
:DDS:OFFSet?
```

Description

Set or query the offset of the signal source signal.

Parameter

Name	Type	Range	Default
<offset>	real	-3.5V-+3.5V	0

Remarks

N/A.

Return Format

Query returns ON,OFF.

Example

```
:DDS:OFFSet 0.5      /*Set the signal source waveform offset to 0.5V */
:DDS:OFFSet?         /* Query returns 5.00000e-01*/
```

3.14.6 :DDS:DUTY

Syntax

```
:DDS:DUTY <duty>
:DDS:DUTY?
```

Description

Set or query the duty cycle of the signal source signal.

Parameter

Name	Type	Range	Default
<duty>	Real	0-99	50

Remarks

N/A

Return Format

Query to return the duty cycle value.

Example

```
:DDS:DUTY 50          /*Set the signal source signal duty cycle to 50%**/
:DDS:DUTY?            /* Query returns 50*/
```

3.14.7 :DDS:WAVE:MODE

Syntax

```
:DDS:WAVE:MODE <bool>
:DDS:WAVE:MODE?
```

Description

Set or query the signal source modulation status.

Parameter

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

Remarks

N/A.

Return Format

The query returns ON, OFF.

Example

```
:DDS:WAVE:MODE ON      /*Turn on the signal source modulation */
:DDS:WAVE:MODE?        /* Query returns ON*/
```

3.14.8 :DDS:MODE:TYPE

Syntax

:DDS:MODE:TYPE <type>

:DDS:MODE:TYPE?

Description

Set or query the signal source modulation type.

Parameter

Name	Type	Range	Default
<type>	Real	{AM FM}	AM

Remarks

N/A.

Return Format

Query returns AM,FM.

Example

:DDS:MODE:TYPE AM /* Set the signal source modulation type to AM */

:DDS:MODE:TYPE? /* Query returns AM*/

3.14.9 :DDS:MODE:WAVE:TYPE

Syntax

:DDS:MODE:WAVE:TYPE <type>

:DDS:MODE:WAVE:TYPE?

Description

Set or query the modulation wave type when setting up signal source modulation.

Parameter

Name	Type	Range	Default
<type>	Real	{SINE SQUAre RAMP}	SINE

Remarks

N/A

Return Format

Query returns SINE, SQUARE, and RAMP.

Example

```
:DDS:MODE:WAVE:TYPE SINE /*Set the signal source modulation wave type to SINE*/
:DDS:MODE:WAVE:TYPE?          /* Query returns SINE */
```

3.14.10 :DDS:MODE:FREQ**Syntax**

```
:DDS:MODE:FREQ <freq>
:DDS:MODE:FREQ?
```

Description

Set or query the frequency of the modulated wave when setting or querying signal source modulation.

Parameter

Name	Type	Range	Default
<freq>	real	0-50KHz	100Hz

Remarks

N/A.

Return Format

Query to return frequency in scientific notation.

Example

```
:DDS:MODE:FREQ 1000          /*Set the signal source modulation wave frequency to
1KHz*/
:DDS:MODE:FREQ?              /* Query returns 1.00000e+03*/
```

3.14.11 :DDS:MODE:DEPTHordeviation**Syntax**

```
:DDS:MODE:DEPTHordeviation <value>
:DDS:MODE:DEPTHordeviation?
```

Description

Set or query the deviation or depth of signal source modulation.

Parameter

N/A

Remarks

N/A.

Return Format

When the modulation type is AM, querying returns the modulation depth value.

When the modulation type is FM, querying returns the deviation.

Example

```
:DDS:MODE:DEPTHordeviation 50      /*Set the modulation depth to 50*/
```

```
:DDS:MODE:DEPTHordeviation?        /* Query returns 50 */
```

3.14.12 :DDS:BURSt:SWITCh

Syntax

```
:DDS:BURSt:SWITCh <bool>
```

```
:DDS:BURSt:SWITCh?
```

Description

Set or query the signal source burst status.

Parameter

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

Remarks

N/A.

Return Format

Query returns ON,OFF.

Example

```
:DDS:BURSt:SWITCh ON      /*Turn on the signal source burst*/
```

```
:DDS:BURSt:SWITCh?        /* Query returns ON*/
```

3.14.13 :DDS:BURSt:TYPE

Syntax

```
:DDS:BURSt:TYPE <type>
```

```
:DDS:BURSt:TYPE?
```

Description

Set or query the signal source burst type.

Parameter

Name	Type	Range	Default
<type>	Real	{N_CYCLE INFin}	N_CYCLE

Remarks

N/A.

Return Format

Query returns N_CYCLE, INFin.

Example

```
:DDS:BURSt:TYPE N_CYCLE      /*Set the burst type to N_CYCLE*/
:DDS:BURSt:TYPE?             /* Query returns N_CYCLE */
```

3.14.14 :DDS:BURSt:CNT

Syntax

```
:DDS:BURSt:CNT <CNT>
:DDS:BURSt:CNT?
```

Description

Set or query the number of signal source signal cycles.

Parameter

Name	Type	Range	Default
<cnt>	Real	0-1024	0

Remarks

N/A.

Return Format

Query returns an integer.

Example

```
:DDS:BURSt:CNT 2             /*Set the number of cycles to 2 */
:DDS:BURSt:CNT?              /* Query returns 2*/
```

3.14.15 :DDS:BURSt:TRIGger

Syntax

:DDS:BURSt:TRIGger

Description

Send a signal source burst once.

Parameter

N/A

Remarks

N/A

Return Format

N/A

Example

N/A

3.14.16 :DDS:ARB:DAC16:BIN

Syntax

:DDS:ARB:DAC16:BIN

Description

Download waveform data. <binary_block_data> indicates binary data to be downloaded. <binary_block_data> is a binary data block starting with the # symbol, such as \"#508192\" binary data. The \"5\" after the # indicates that 5 characters represent the data length information, and \"8192\" indicates that there are 8192 bytes of binary data. Each waveform point corresponds to a 2-byte binary number (for example, point 1024 corresponds to the binary number 0x0400; since the low byte comes first and the high byte comes after, it is 0004). Therefore, the number of bytes must be even. Note: The number of arbitrary wave waveform points must be 4096.

Parameter

N/A

Remarks

N/A

Return Format

N/A

Example

N/A

3.15 :AUToscale

Syntax

:AUToscale

Description

Enable the waveform auto-setup function. The oscilloscope will automatically adjust the vertical scale, horizontal time base, and trigger mode according to the input signal to achieve the optimal waveform display. This command function is equivalent to pressing the AUTO Scale button on the front panel.

Parameter

N/A

Remarks

- When using the waveform auto-setup function, for sine signals, the theoretical frequency is not less than 10Hz; for square wave signals, the theoretical value is related to the duty cycle, which should be greater than 1%, and the amplitude should be at least 10mVpp (when the probe ratio is 1X).
- When the Pass/Fail function is in the allowed test state (see: MASK:ENABle command), if this command is sent, the oscilloscope first disables the Pass/Fail function and then executes the waveform auto-setup function.
- This command is invalid when the waveform recording function is enabled or when replaying recorded waveforms.

Return Format

N/A

Example

:AUToscale /*The oscilloscope performs the AUTO function*/

3.16 :RUNing

Syntax

:RUNing

Description

The command starts/ stops the oscilloscope.

Parameter

N/A

Remarks

N/A

Return Format

N/A

Example

:RUNing ON /*The oscilloscope starts running*/

3.17 :SINGle

Syntax

:SINGle

Description

Set the oscilloscope to single trigger mode. This command functions equivalently to pressing the front-panel [Single] button.

Parameter

N/A

Remarks

- In single-trigger mode, the oscilloscope will trigger once when the trigger condition is met and then stop.
- This command is invalid when the waveform recording function is enabled or when replaying recorded waveforms.

Return Format

N/A

Example

```
:SINGle      /* Oscilloscope single acquisition*/
```

3.18 :BUS Commands

3.18.1 :BUS:CAN:SOURce

Syntax

```
:BUS:CAN:SOURce <source>
```

```
:BUS:CAN:SOURce?
```

Description

Set up or query measurement sources.

Parameter

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

Remarks

N/A

Return Format

Query returns CHAN1,CHAN2.

Example

```
:BUS:CAN:SOURce CHANnel2      /* Set the measurement source to CH2*/
```

```
:BUS:CAN:SOURce?              /* Query returns CHAN2*/
```

3.18.2 :BUS:CAN:THReshold

Syntax

```
:BUS:CAN:THReshold <source>
```

```
:BUS:CAN:THReshold?
```

Description

Set or query channel threshold thresholds.

Parameter

Name	Type	Range	Default
<source>	float	-1.0V~1.0V, unit: V	default 0

Remarks

N/A

Return Format

Query the threshold threshold for returning the current operation result in scientific counting form.

Example

```
:BUS:CAN:THReshold 1      /* Set the channel threshold threshold to 1V*/
:BUS:CAN:THReshold?        /* Query returns 1.000000 e+00*/
```

3.18.3 :BUS:CAN:SType

Syntax

```
:BUS:CAN:SType <type>
:BUS:CAN:SType?
```

Description

Set or query the signal type.

Parameter

Name	Type	Range	Default
<type>	Discrete	{CANH CANL RX TX DIFF}	CANH

Remarks

N/A

Return Format

Query returns CANH, CANL, RX, TX, or DIFF.

Example

```
:BUS:CAN:SType CANH      /* Set CAN signal type to CANH */
:BUS:CAN:SType?          /* Query returns CANH */
```

3.18.4 :BUS:CAN:BAUD

Syntax

:BUS:CAN:BAUD <type>

:BUS:CAN:BAUD?

Description

Set or query signal rate.

Parameter

Name	Type	Range	Default
<type>	Discrete	{10000 20000 33000 50000 62500 83300 100000 125000 250000 500000 800000 1000000 2000000 4000000 5000000}	10000

Remarks

N/A

Return Format

Query returns

10kbps,20kbps,33.3kbps,50kbps,62.5kbps,83.3kbps,100kbps,125kbps,250kbps,500kbps,800kbps,1Mbps,2Mbps,4Mbps,5Mbps.

Example

:BUS:CAN:BAUD 10000 /* Set the signal rate to 10kbps */

:BUS:CAN:BAUD? /* Query returns 10kbps */

3.18.5 :BUS:CAN:SPOint

Syntax

:BUS:CAN:SPOint <spoint>

:BUS:CAN:SPOint?

Description

Set or query the sampling point location.

Parameter

Name	Type	Range	Default
<spoint>	Integer	10-90	10

Remarks

N/A

Return Format

The query returns 10 to 90 (expressed as a percentage).

Example

```
:BUS:CAN:SPOint 10      /* Set the sampling point position to 10*/
:BUS:CAN:SPOint?        /* Query returns 10*/
```

3.18.6 :BUS:IIC:SCLK:SOURce

Syntax

```
:BUS:IIC:SCLK:SOURce <source>
:BUS:IIC:SCLK:SOURce?
```

Description

Set or query the clock signal source.

Parameter

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

Remarks

N/A

Return Format

Query returns CHAN1,CHAN2.

Example

```
:BUS:IIC:SCLK:SOURce CH2      /* Set the clock signal source to CH2*/
:BUS:IIC:SCLK:SOURce?        /* Query returns CH2*/
```

3.18.7 :BUS:IIC:SDA:SOURce

Syntax

```
:BUS:IIC:SDA:SOURce <source>
:BUS:IIC:SDA:SOURce?
```

Description

Set or query data sources.

Parameter

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

Remarks

N/A

Return Format

Query returns CHAN1,CHAN2.

Example

:BUS:IIC:SDA:SOURce CHANnel2 /* Set the data source to CH2*/

:BUS:IIC:SDA:SOURce? /* Query returns CH2*/

3.18.8 :BUS:IIC:SCLK:THReshold

Syntax

:BUS:IIC:SCLK:THReshold <source>

:BUS:IIC:SCLK:THReshold?

Description

Set or query the clock threshold threshold.

Parameter

Name	Type	Range	Default
<source>	float	-1.0V~1.0V, unit: V	default 0

Remarks

N/A

Return Format

Query the threshold threshold for returning the current operation result in scientific counting form.

Example

:BUS:IIC:SCLK:THReshold 1 /* Set the clock threshold to 1V */

:BUS:IIC:SCLK:THReshold? /* Query returns 1.000000 e+00*/

3.18.9 :BUS:IIC:SDA:THReshold

Syntax

:BUS:IIC:SDA:THReshold <source>

:BUS:IIC:SDA:THReshold?

Description

Set or query data threshold thresholds.

Parameter

Name	Type	Range	Default
<source>	float	-1.0V~1.0V, unit: V	default 0

Remarks

N/A

Return Format

Query the threshold threshold for returning the current operation result in scientific counting form.

Example

```
:BUS:IIC:SDA:THReshold 1      /* Set the data threshold threshold to 1V */
:BUS:IIC:SDA:THReshold?      /* Query returns 1.000000 e+00*/
```

3.18.10 :BUS:IIC:ADDRess

Syntax

```
:BUS:IIC:ADDRess <source>
:BUS:IIC:ADDRess?
```

Description

Set or query address mode.

Parameter

Name	Type	Range	Default
<source>	Discrete	{RW NORW}	RW

Remarks

NORW: Address bit width does not include R/W bits.

Return Format

Query returns RW or NORW.

Example

```
:BUS:IIC:ADDRess NORW      /* Set the address mode to NORW */
```

:BUS:IIC:ADDRess? /* Query returns NORW */

3.18.11 :BUS:LIN:SOURce

Syntax

:BUS:LIN:SOURce <source>

:BUS:LIN:SOURce?

Description

Setting or querying LIN sources.

Parameter

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

Remarks

N/A

Return Format

Query returns CHAN1,CHAN2.

Example

```
:BUS:LIN:SOURce CHANnel2 /* Set the measurement source to CH2*/
:BUS:LIN:SOURce? /* Query returns CHAN2*/
```

3.18.12 :BUS:LIN:THReshold

Syntax

:BUS:LIN:THReshold <source>

:BUS:LIN:THReshold?

Description

Set or query channel threshold thresholds.

Parameter

Name	Type	Range	Default
<source>	float	-1.0V~1.0V, unit: V	default 0

Remarks

N/A

Return Format

Query the threshold threshold for returning the current operation result in scientific counting form.

Example

```
:BUS:LIN:THReshold 1      /* Set the channel threshold threshold to 1V */
:BUS:LIN:THReshold?       /* Query returns 1.000000 e+00*/
```

3.18.13 :BUS:LIN:BAUD

Syntax

```
:BUS:LIN:BAUD <type>
:BUS:LIN:BAUD?
```

Description

Set or query signal rate.

Parameter

Name	Type	Range	Default
<type>	Discrete	{2400 4800 9600 19200 38400 57600 115200 230400 460800 921600 1000000 2000000 5000000 10000000 20000000}	115200

Remarks

N/A

Return Format

Query returns
2.4kbps,4.8kbps,9.6kbps,19.2kbps,38.4kbps,57.6kbps,115.2kbps,230.4kbps,460.8kbps,921.6kbps,1M,2M,5M,10M,20M.

Example

```
:BUS:LIN:BAUD 2400      /* Set the signal rate to 2.4kbps*/
:BUS:LIN:BAUD?          /* Query returns 2.4kbps*/
```

3.18.14 :BUS:LIN:PARity

Syntax

:BUS:LIN:PARity <type>

:BUS:LIN:PARity?

Description

Set or query the checksum under LIN.

Parameter

Name	Type	Range	Default
<type>	Discrete	{WITHOUT WITH}	WITHOUT

Remarks

N/A

Return Format

Query returns either THE or THE.

Example

:BUS:LIN:PARity WITHOUT /* Set the LIN checksum to WITOUT */

:BUS:LIN:PARity? /* Query returns with OUT */

3.18.15 :BUS:LIN:STANdard

Syntax

:BUS:LIN:STANdard <type>

:BUS:LIN:STANdard?

Description

Set or query the LIN version.

Parameter

Name	Type	Range	Default
<type>	Discrete	{1.X 2.X BOTH}	1.X

Remarks

N/A

Return Format

Query returns 1.X, 2.X, or BOTH.

Example

:BUS:LIN:STANdard 1 /* Set LIN version to 1.X */

:BUS:LIN:STANdard? /* Query returns 1.X */

3.18.16 :BUS:RS232:BAUD

Syntax

:BUS:RS232:BAUD <baud>

:BUS:RS232:BAUD?

Description

Set or query the baud rate.

Parameter

Name	Type	Range	Default
<baud>	Discrete	{600 1200 1800 2000 2400 4800 9600 19200 38400 57600 115200 230400 460800 921600 1000000 2000000 5000000 10000000 20000000}	115200

Remarks

N/A

Return Format

Query returns

600bps, 1.2kbps, 1.8kbps, 2.0kbps, 2.4kbps, 4.8kbps, 9.6kbps, 19.2kbps, 38.4 kbps, 57.6 kbps, 115.2 kbps, 230.4 kbps, 460.8 kbps, 921.6 kbps, 1M, 2M, 5M, 10M or 20M.

Example

:BUS:RS232:BAUD 2000 /* Set the baud rate to 2000*/

:BUS:RS232:BAUD? /*Query returns 2.0kbps*/

3.18.17 :BUS:RS232:TX

Syntax

:BUS:RS232:TX <source>

:BUS:RS232:TX?

Description

Set or query TX measurement sources.

Parameter

Name	Type	Range	Default
------	------	-------	---------

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

Remarks

N/A

Return Format

Query returns CHAN1,CHAN2.

Example

```
:BUS:RS232:TX CHANnel2      /* Set the measurement source to CH2.*/
:BUS:RS232:TX?              /* Query returns CHAN2*/
```

3.18.18 :BUS:RS232:TX:THReshold

Syntax

```
:BUS:RS232:TX:THReshold <source>
:BUS:RS232:TX:THReshold?
```

Description

Set or query the channel TX threshold threshold.

Parameter

Name	Type	Range	Default
<source>	float	-1.0V~1.0V, unit: V	default 0

Remarks

N/A

Return Format

Query the threshold threshold for returning the current operation result in scientific counting form.

Example

```
:BUS:RS232:TX:THReshold 1      /* Set the channel threshold threshold to 1V */
:BUS:RS232:TX:THReshold?       /* Query returns 1.000000 e+00*/
```

3.18.19 :BUS:RS232:RX

Syntax

:BUS:RS232:RX <source>

:BUS:RS232:RX?

Description

Set or query RX measurement sources.

Parameter

Name	Type	Range	Default
<source>	Discrete	{CHANnel1 CHANnel2}	OFF

Remarks

N/A

Return Format

Query returns CHAN1,CHAN2.

Example

:BUS:RS232:RX CHANnel2 /* Set the measurement source to CH2*/

:BUS:RS232:RX? /* Query returns CHAN2*/

3.18.20 :BUS:RS232:RX:THReshold

Syntax

:BUS:RS232:RX:THReshold <source>

:BUS:RS232:RX:THReshold?

Description

Set or query channel RX threshold threshold.

Parameter

Name	Type	Range	Default
<source>	float	-1.0V~1.0V, unit: V	default 0

Remarks

N/A

Return Format

Query the threshold threshold for returning the current operation result in scientific counting form.

Example

```
:BUS:RS232:RX:THReshold 1      /* Set the channel threshold threshold to 1V */
:BUS:RS232:RX:THReshold?        /*Query returns 1.000000e+00*/
```

3.18.21 :BUS:RS232:POLarity

Syntax

```
:BUS:RS232:POLarity <polarity>
:BUS:RS232:POLarity?
```

Description

Set or query RS232 polarity.

Parameter

Name	Type	Range	Default
<source>	Discrete	{NEGative POSitive}	POSitive

Remarks

N/A

Return Format

Query returns NEG or POS.

Example

```
:BUS:RS232:POLarity POSitive    /* Set RS232 polarity to positive polarity */
:BUS:RS232:POLarity?            /*Query returns POS*/
```

3.18.22 :BUS:RS232:ENDian

Syntax

```
:BUS:RS232:ENDian <endian>
:BUS:RS232:ENDian?
```

Description

Set or query data transmission bit order.

Parameter

Name	Type	Range	Default
<endian>	Discrete	{LSB MSB}	MSB

Remarks

N/A

Return Format

Query returns LSB or MSB.

Example

```
:BUS:RS232:ENDian LSB      /* Set the data transmission bit sequence to LSB */
:BUS:RS232:ENDian?          /*Query returns LSB*/
```

3.18.23 :BUS:RS232:DBITs

Syntax

```
:BUS:RS232:DBITs <value>
:BUS:RS232:DBITs?
```

Description

Set or query data bit width.

Parameter

Name	Type	Range	Default
<value>	Integer	{5 6 7 8 9}	8

Remarks

N/A

Return Format

Query returns 5,6,7,8 or 9.

Example

```
:BUS:RS232:DBITs 5          /* Set the data bit width to 5*/
:BUS:RS232:DBITs?           /*Query returns 5*/
```

3.18.24 :BUS:RS232:PARity

Syntax

```
:BUS:RS232:PARity <value>
:BUS:RS232:PARity?
```

Description

设置 or 查询 RS232 校验方式.

Parameter

Name	Type	Range	Default
< value >	Discrete	{NONE ODD EVENT}	NONE

Remarks

N/A

Return Format

Query returns NONE,ODD or EVENT.

Example

```
:BUS:RS232:PARity ODD      /* Set RS232 verification method to odd verification */
:BUS:RS232:PARity?         /*Query returns ODD*/
```

3.18.25 :BUS:RS232:STOP

Syntax

```
:BUS:RS232:STOP <value>
:BUS:RS232:STOP?
```

Description

Set or query RS232 stop bit.

Parameter

Name	Type	Range	Default
<value>	Discrete	{1 1.5 2}	1

Remarks

N/A

Return Format

Query returns 1,1.5 or 2.

Example

```
:BUS:RS232:STOP 1          /* Set the stop digit to 1 */
:BUS:RS232:STOP?           /*Query returns 1*/
```

3.18.26 :BUS:SPI:SCLK:SOURce

Syntax

:BUS:SPI:SCLK:SOURce <source>

:BUS:SPI:SCLK:SOURce?

Description

Set or query SPI clock source.

Parameter

Name	Type	Range	Default
<source>	Discrete	{CHANnel1 CHANnel2}	OFF

Remarks

N/A

Return Format

Query returns CHAN1,CHAN2.

Example

:BUS:SPI:SCLK:SOURce CHANnel2 /* Set SPI clock signal source to CH2*/

:BUS:SPI:SCLK:SOURce? /*Query returns CHAN2*/

3.18.27 :BUS:SPI:SCLK:SLOPe

Syntax

:BUS:SPI:SCLK:SLOPe <source>

:BUS:SPI:SCLK:SLOPe?

Description

Set or query SPI clock edge type.

Parameter

Name	Type	Range	Default
<source>	Discrete	{POSitive NEGative}	POSitive

Remarks

N/A

Return Format

Query returns POS or NEG.

Example

```
:BUS:SPI:SCLK:SLOPe POSitive /*设 Set SPI clock edge type to Positive */
:BUS:SPI:SCLK:SLOPe? /*Query returns POS*/
```

3.18.28 :BUS:SPI:MISO:SOURce**Syntax**

```
:BUS:SPI:MISO:SOURce <source>
:BUS:SPI:MISO:SOURce?
```

Description

Set or query the MISO source type.

Parameter

Name	Type	Range	Default
<source>	Discrete	{CHANnel1 CHANnel2}	OFF

Remarks

N/A

Return Format

Query returns CHAN1,CHAN2.

Example

```
:BUS:SPI:MISO:SOURce CHANnel2 /* Set MISO source to CH2*/
:BUS:SPI:MISO:SOURce? /*Query returns CHAN2*/
```

3.18.29 :BUS:SPI:MOSI:SOURce**Syntax**

```
:BUS:SPI:MOSI:SOURce <source>
:BUS:SPI:MOSI:SOURce?
```

Description

Set or query MOSI source type.

Parameter

Name	Type	Range	Default
<source>	Discrete	{CHANnel1 CHANnel2}	OFF

Remarks

N/A

Return Format

Query returns CHAN1,CHAN2.

Example

:BUS:SPI:MOSI:SOURce CHANnel2 /* Set the MOSI signal source to CH2*/

:BUS:SPI:MOSI:SOURce? /*Query returns CHAN2*/

3.18.30 :BUS:SPI:SCLK:SOURce:THReshold

Syntax

:BUS:SPI:SCLK:SOURce:THReshold <source>

:BUS:SPI:SCLK:SOURce:THReshold?

Description

Set or query clock threshold threshold.

Parameter

Name	Type	Range	Default
<source>	float	-1.0V~1.0V, unit: V	default 0

Remarks

N/A

Return Format

Query the threshold threshold for returning the current operation result in scientific counting form.

Example

:BUS:SPI:SCLK:SOURce:THReshold 1 /* Set the clock threshold to 1V */

:BUS:SPI:SCLK:SOURce:THReshold? /*Query returns 1.000000e+00*/

3.18.31 :BUS:SPI:MISO:SOURce:THReshold

Syntax

:BUS:SPI:MISO:SOURce:THReshold <source>

:BUS:SPI:MISO:SOURce:THReshold?

Description

Set or query MISO threshold threshold.

Parameter

Name	Type	Range	Default
<value>	float	-1.0V~1.0V, unit: V	default 0

Remarks

N/A

Return Format

Query the threshold threshold for returning the current operation result in scientific counting form.

Example

```
:BUS:SPI:MISO:SOURce:THReshold 1      /* Set the MISO threshold to 1V */
:BUS:SPI:MISO:SOURce:THReshold?        /*Query returns 1.000000e+00*/
```

3.18.32 **:BUS:SPI:MOSI:SOURce:THReshold**

Syntax

```
:BUS:SPI:MOSI:SOURce:THReshold <source>
:BUS:SPI:MOSI:SOURce:THReshold?
```

Description

Set or query MOSI threshold threshold.

Parameter

Name	Type	Range	Default
<source>	float	-1.0V~1.0V, unit: V	default 0

Remarks

N/A

Return Format

Query the threshold threshold for returning the current operation result in scientific counting form.

Example

```
:BUS:SPI:MOSI:SOURce:THReshold 1      /* Set the MOSI threshold to 1V */
:BUS:SPI:MOSI:SOURce:THReshold?        /*Query returns 1.000000e+00*/
```

3.18.33 :BUS:SPI:MODE

Syntax

```
:BUS:SPI:MODE <mode>
:BUS:SPI:MODE?
```

Description

Set or query SPI decoding mode.

Parameter

Name	Type	Range	Default
<mode>	Discrete	{TIMEOUT CS}	TIMEOUT

Remarks

N/A

Return Format

Query returns TIMEOUT or CS.

Example

```
:BUS:SPI:MODE TIMEOUT      /* Set SPI decoding mode to timeout */
:BUS:SPI:MODE?              /*Query returns TIMEOUT */
```

3.18.34 :BUS:SPI:CS:SOURce

Syntax

```
:BUS:SPI:CS:SOURce <source>
:BUS:SPI:CS:SOURce?
```

Description

Set or query CS film selection source.

Parameter

Name	Type	Range	Default
<source>	Discrete	{CHANnel1 CHANnel2}	OFF

Remarks

N/A

Return Format

Query returns CHAN1,CHAN2.

Example

```
:BUS:SPI:CS:SOURce CHANnel2      /* Set CS film selection source to CH2 */
:BUS:SPI:CS:SOURce?              /*Query returns CHAN2*/
```

3.18.35 :BUS:SPI:CS:POLarity

Syntax

```
:BUS:SPI:CS:POLarity <polarity>
:BUS:SPI:CS:POLarity?
```

Description

Set or query the polarity of the film selection line.

Parameter

Name	Type	Range	Default
<polarity>	Discrete	{POSitive NEGative}	POS

Remarks

N/A

Return Format

Query returns POS or NEG.

Example

```
:BUS:SPI:CS:POLarity NEGative    /* Set the polarity of the chip selection line to
negative polarity */
:BUS:SPI:CS:POLarity?            /*Query returns NEG*/
```

3.18.36 :BUS:SPI:CS:THReshold

Syntax

```
:BUS:SPI:CS:THReshold <value>
:BUS:SPI:CS:THReshold?
```

Description

Set or query the threshold threshold for line selection.

Parameter

Name	Type	Range	Default
<value>	float	-1.0V~1.0V, unit: V	default 0

Remarks

N/A

Return Format

Query the threshold threshold for returning the current operation result in scientific counting form.

Example

```
:BUS:SPI:CS:THReshold 1 /* Set the threshold threshold for line selection to 1V */
:BUS:SPI:CS:THReshold? /*Query returns 1.000000e+00*/
```

3.18.37 :BUS:SPI:ENDian

Syntax

```
:BUS:SPI:ENDian <endian>
:BUS:SPI:ENDian?
```

Description

Set or query the bit order of SPI.

Parameter

Name	Type	Range	Default
<endian>	Discrete	{LSB MSB}	MSB

Remarks

N/A

Return Format

Query returns LSB or MSB.

Example

```
:BUS:SPI:ENDian LSB /* Set the bit order of the chip selection line to LSB */
:BUS:SPI:ENDian? /*Query returns LSB*/
```

3.18.38 :BUS:SPI:DBITs

Syntax

:BUS:SPI:DBITs <value>

:BUS:SPI:DBITs?

Description

Set or query SPI data bit width.

Parameter

Name	Type	Range	Default
<value>	Discrete	4-32	8

Remarks

N/A

Return Format

Query returns 4,8,16,32,Custom.

Example

:BUS:SPI:DBITs 4 /* Set the data bit width to 4 */

:BUS:SPI:DBITs? /*Query returns 4*/

3.18.39 :BUS:SPI:POLarity

Syntax

:BUS:SPI:POLarity <polarity>

:BUS:SPI:POLarity?

Description

Set or query SPI polarity.

Parameter

Name	Type	Range	Default
<polarity>	Discrete	{POSitive NEGative}	POSitive

Remarks

N/A

Return Format

Query returns POS or NEG.

Example

```
:BUS:SPI:POLarity POSitive      /* Set SPI polarity to positive polarity */  
:BUS:SPI:POLarity?              /*Query returns POS*/
```



Addr: #35 Building, No. 780 Baoyuan Road, High-tech Zone, Qingdao, Shandong, China 266114

Switchboard: 400-036-7077

Email: service@hantek.com

Tel: (0086)532-55678770 & 55678772 & 55678773

Zip code: 266114

Website: www.hantek.com

Qingdao Hantek Electronics Co., LTD