Hantek®

SCPI Programmers Manual

DSO2000 Series Digital Storage Oscilloscope

Introduction to SCPI commands

SCPI is the abbreviation of Programmable Instrument Standard Commands. It defines a set of standard syntax and commands for controlling programmable test and measurement instruments. SCPI commands are ASCII strings that are passed into the instrument through the physical transport layer. The command consists of a series of keywords, and some also need to include parameters. In the protocol, the command is specified as the following form: CONFigure. In use, you can either write the full name or only the abbreviation that contains only uppercase letters. Usually, the feedback of the instrument to the query command is also an ASCII code.

Command format

Commands usually begin with a colon ":". Keywords are separated by a colon ":", and keywords are followed by optional parameter settings. A question mark "?" is added after the command line to indicate that this function is being queried. The command keyword and the first parameter are separated by a space.
For example:
:ACQuire:TYPE
:ACQuire:TYPE?
ACQuire is the root keyword of the command, and TYPE is the second-level keyword. The command line starts with a colon ":", and the keywords of each level are separated with a colon ":". indicates a parameter that can be set. The question mark "?" Indicates a query. Command keyword: ACQuire: TYPE and parameter are separated by spaces.
In some commands with parameters, multiple parameters are usually separated by a comma ",", for example:
[:TRACe[<n>]]:DATA:VALue volatile,<points>,<data>

Symbol Description

The following symbols are not sent with the command.
1. Braces {}
The contents of the braces are parameter options. Parameter items are usually separated by a vertical bar "|". When using a command, you must select one of the parameters.
2. Vertical bar |
Vertical bar is used to separate multiple parameter options. One of the parameters must be selected when using the command.
3. Square brackets []
The content in square brackets is omissible.
4. Triangle brackets <>
The parameters in the triangle brackets must be replaced with a valid value.

Command abbreviation

All commands are not case sensitive, you can use all uppercase or lowercase. But if you want to abbreviate, you must enter all capital letters in the command format.

For example:

:MEASure:ADISplay? Can be abbreviated to :MEAS:ADIS?

Contents

Introduction to SCPI commands 2
1. CHANnel<n> Command Subsystem 9
- 1.1 CHANnel<n>:BWLimit 9
- 1.2 CHANnel<n>:COUPling 9
- 1.3 CHANnel<n>:DISPlay 10
- 1.4 CHANnel<n>:INVert 10
- 1.5 CHANnel<n>:OFFSet 10
- 1.6 CHANnel<n>:SCALe 11
- 1.7 CHANnel<n>:PROBe 11
- 1.8 CHANnel<n>:VERNier 11

2. TIMebase Command Subsystem 12
- 2.1 TIMebase:WINDow:ENABle 12
- 2.2 TIMebase:WINDow:POSition 12
- 2.3 TIMebase:WINDow:SCALe 13
- 2.4 TIMebase:POSition 13
- 2.5 TIMebase:SCALe 13
- 2.6 TIMebase:RANGe 13
- 2.7 TIMebase:MODE 13

3. ACQuire Command Subsystem 14
- 3.1 ACQuire:POINts 14
- 3.2 ACQuire:TYPE 14
- 3.3 ACQuire:SRATe? 15
- 3.4 ACQuire:COUNt 15

4. TRIGger Command Subsystem 15
- 4.1 TRIGger:FORCe 16
- 4.2 TRIGger:MODE 16
- 4.3 TRIGger:STATus? 16
- 4.4 TRIGger:SWEep 17
- 4.5 TRIGger:HOLDoff 17
- 4.6 TRIGger:EDGe 17
  - 4.6.1 TRIGger:EDGe:SOURce 17
  - 4.6.2 TRIGger:EDGe:SLOPe 18
  - 4.6.3 TRIGger:EDGe:LEVel 18
- 4.7 TRIGger:PULSe 18
  - 4.7.1 TRIGger:PULSe:SOURce 18
  - 4.7.2 TRIGger:PULSe:POLarity 18
  - 4.7.3 TRIGger:PULSe:WHEN 19
  - 4.7.4 TRIGger:PULSe:WIDth 19
  - 4.7.5 TRIGger:PULSe:LEVel 19
- 4.8 TRIGger:SLOPe 19

4.8.1 TRIGger:SLOPe:SOURce 20
4.8.2 TRIGger:SLOPe:POLarity 20
4.8.3 TRIGger:SLOPe:WHEN 20
4.8.4 TRIGger:SLOPe:WIDth 21
4.8.5 TRIGger:SLOPe:ALEVel 21
4.8.6 TRIGger:SLOPe:BLEVel 21
4.9 TRIGger:TV 21
4.9.1 TRIGger:TV:SOURce 22
4.9.2 TRIGger:TV:POLarity 22
4.9.3 TRIGger:TV:MODE 22
4.9.4 TRIGger:TV:LINE 22
4.9.5 TRIGger:TV:STANdard 23
4.9.6 TRIGger:VIDeo:LEVel 23
4.10 TRIGger:TIMeout 23
4.10.1 TRIGger:TIMeout:SOURce 23
4.10.2 TRIGger:TIMeout:LEVel 23
4.10.3 TRIGger:TIMeout:WIDth 24
4.10.4 TRIGger:TIMeout:POLarity 24
4.11 TRIGger:WINDOw 24
4.11.1 TRIGger:WINDOw:SOURce 24
4.11.2 TRIGger:WINDOw:ALEVel 24
4.11.3 TRIGger:WINDOw:BLEVel 25
4.12 TRIGger:INTERVAl 25
4.12.1 TRIGger:INTERVAl:SOURce 25
4.12.2 TRIGger:INTERVAl:SLOp 25
4.12.3 TRIGger:INTERVAl:WHEN 26
4.12.4 TRIGger:INTERVAl:TIME 26
4.12.5 TRIGger:INTERVAl:ALEVel 26
4.13 TRIGger:UNDER_Am 26
4.13.1 TRIGger:UNDER_Am:SOURce 27
4.13.2 TRIGger:UNDER_Am:POLarity 27
4.13.3 TRIGger:UNDER_Am:WHEN 27
4.13.4 TRIGger:UNDER_Am:TIME 27
4.13.5 TRIGger:UNDER_Am:ALEVel 27
4.13.6 TRIGger:UNDER_Am:BLEVel 28
4.14 TRIGger:UART 28
4.14.1 TRIGger:UART:SOURce 28
4.14.2 TRIGger:UART:CONdition 28
4.14.3 TRIGger:UART:BAUd 28
4.14.4 TRIGger:UART:ALEVel 29
4.14.5 TRIGger:UART:DATA 29
4.14.6 TRIGger:UART:WIDTh 29

4.14.7 TRIGger:UART:PARIty 29

4.15 TRIGger:CAN 30

4.15.1 TRIGger:CAN:SOURce 30

4.15.2 TRIGger:CAN:IDLe 30

4.15.3 TRIGger:CAN:BAUd 30

4.15.4 TRIGger:CAN:CONdition 31

4.15.5 TRIGger:CAN:ID 31

4.15.6 TRIGger:CAN:DLC 31

4.15.7 TRIGger:CAN:DATA 31

4.15.8 TRIGger:CAN:ALEVel 31

4.16 TRIGger:LIN 32

4.16.1 TRIGger:LIN:SOURce 32

4.16.2 TRIGger:LIN:IDLe 32

4.16.3 TRIGger:LIN:BAUd 32

4.16.4 TRIGger:LIN:CONdition 33

4.16.5 TRIGger:LIN:ID 33

4.16.6 TRIGger:LIN:DATA 33

4.16.7 TRIGger:LIN:ALEVel 33

4.17 TRIGger:IIC 33

4.17.1 TRIGger:IIC:SDA:SOURce 34

4.17.2 TRIGger:IIC:SCL:SOURce 34

4.17.3 TRIGger:IIC:CONdition 34

4.17.4 TRIGger:IIC:ADDer 34

4.17.5 TRIGger:IIC:DATA 35

4.17.6 TRIGger:IIC:ALEVel 35

4.17.7 TRIGger:IIC:BLEVel 35

4.18 TRIGger:SPI 35

4.18.1 TRIGger:SPI:SDA:SOURce 36

4.18.2 TRIGger:SPI:SCL:SOURce 36

4.18.3 TRIGger:SPI:SCK 36

4.18.4 TRIGger:SPI:WIDth 36

4.18.5 TRIGger:SPI:DATA 36

4.18.6 TRIGger:SPI:MASK 37

4.18.7 TRIGger:SPI:ALEVel 37

4.18.8 TRIGger:SPI:BLEVel 37

4.19 TRIGger:PATTern 37

4.19.1 TRIGger:PATTern:PATTern 37

4.19.2 :TRIGger:PATTern:LEVel 38

:TRIGger:PATTern:LEVel <chan>,<level> 38

5. CALibrate Command Subsystem 38

5.1 CALibrate:STARt 38

5.2 CALibrate:STATus? 39

5.3 CALibrate:QUIT....39
6. MATH Command Subsystem....39
6.1 MATH:DISPlay....39
6.2 MATH:OPERator....40
6.3 MATH:SOURce1....40
6.4 MATH:SOURce2....40
6.5 MATH:SCALe....40
6.6 MATH:OFFSet....41
6.7 MATH:FFT:SOURce....41
6.8 MATH:FFT:WINDow....41
6.9 MATH:FFT:UNIT....41
6.10 MATH:FFT:HSCale....41
6.11 MATH:FFT:HCENter....42
7. WAVeform Command Subsystem....42
7.1 WAVeform:DATA:ALL?....42
8. DISPlay Command Subsystem....43
8.1 DISPlay:TYPE....43
8.2 DISPlay:WBRightness....43
8.3 DISPlay:GRID....44
8.4 DISPlay:GBRightness....44
9. CURSor Command Subsystem....44
9.1 CURSor:MODE....45
9.2 CURSor:MANual:TYPE....45
9.3 CURSor:MANual:SOURce....45
9.4 CURSor:MANual:AX....46
9.5 CURSor:MANual:AXValue?....46
9.6 CURSor:MANual:AY....46
9.7 CURSor:MANual:AYValue?....46
9.8 CURSor:MANual:BX....47
9.9 CURSor:MANual:BXValue?....47
9.10 CURSor:MANual:BY....47
9.11 CURSor:MANual:BYValue?....47
9.12 CURSor:TRACk:SOURcea....47
9.13 CURSor:TRACk:SOURceb....48
9.14 CURSor:TRACk:AX....48
9.15 CURSor:TRACk:AXValue?....48
9.16 CURSor:TRACk:AY?....48
9.17 CURSor:TRACk:AYValue?....48
9.18 CURSor:TRACk:BX....49
9.19 CURSor:TRACk:BXValue?....49
9.20 CURSor:TRACk:BY?....49
9.21 CURSor:TRACk:BYValue?....49

10. MEASure Command Subsystem......49
    10.1 MEASure:ENABle......50
    10.2 MEASure:SOURce......50
    10.3 MEASure:ADISplay......50
    10.4 MEASure: CHANnel<n>:ITEM......50
    10.5 MEASure:GATE:ENABle......51
    10.6 MEASure:GATE:AY......51
    10.7 MEASUre:GATE:BY......51
11. MASK Command Subsystem......51
    11.1 MASK:EANBle......52
    11.2 MASK:SOURce......52
    11.3 MASK:MDISplay......52
    11.4 MASK:Output......52
    11.5 MASK:SOOutput......53
    11.6 MASK:X......53
    11.7 MASK:Y......53
    11.8 MASK:CREate......53
12. SYSTem Command Subsystem......53
    12.1 SYSTem:GAM?......54
    12.2 SYSTem:RAM?......54
    12.3 SYSTem:PON......54
    12.4 SYSTem:LANGuage......54
    12.5 SYSTem:LOCKed......54
    12.6 SETUp:ALL?......55
13. DDS Command Subsystem......57
    13.1 DDS:SWITch......57
    13.2 DDS:TYPE......58
    13.3 DDS:FREQ......58
    13.4 DDS:AMP......58
    13.5 DDS:OFFSet......58
    13.6 DDS:DUTY......58
    13.7 DDS:WAVE:MODE......59
    13.8 DDS:MODE:TYPE......59
    13.9 DDS:MODE:WAVE:TYPE......59
    13.10 DDS:MODE:FREQ......59
    13.11 DDS:MODE:DEPThordeviation......60
    13.12 DDS:BURSt:SWITch......60
    13.13 DDS:BURSt:TYPE......60
    13.14 DDS:BURSt:CNT......60
    13.15 DDS:BURSt:TRIGger......60
    13.16 DDS:ARB:DAC16:BIN......61

1. CHANnel<n> Command Subsystem

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

Command List

- **CHANnel<n>:BWLimit**
- **CHANnel<n>:COUPling**
- **CHANnel<n>:DISPlay**
- **CHANnel<n>:INVert**
- **CHANnel<n>:OFFSet**
- **CHANnel<n>:SCALe**
- **CHANnel<n>:PROBe**
- **CHANnel<n>:VERNier**

1.1 CHANnel<n>:BWLimit

Command	:CHANnel<n>:BWLimit <type> :CHANnel<n>:BWLimit?
Description	Set or query the parameters of bandwidth limit for the indicated channel.

Parameters	:<type>::= {{1 | ON} | {0 | OFF}
	:<n>::= {1 | 2 | 3 | 4}

State	▹OFF: Turn off the 20MHz bandwidth limit, and the high-frequency components contained in the measured signal can pass. ▹ON: Turn on the bandwidth limitation, and the high-frequency components contained in the signal under test are attenuated. Turning on bandwidth limiting reduces waveform noise, but attenuates high-frequency components.
Return	The query returns 0 or 1.
Example	:CHANnel1:BWLimit 1 /*Enable 20MHz bandwidth limit*/ :CHANnel1:BWLimit? /*The query returns 1*/

1.2 CHANnel<n>:COUPling

Command	:CHANnel<n>:COUPling <coupling> :CHANnel<n>:COUPling?
Description	Set or query the connection of the indicated channel signal.

Parameters	:<coupling> ::= {AC | DC | GND}
	:<n>::= {1 | 2 | 3 | 4}

State	▹AC: The DC component of the signal under test is blocked. ▹DC: Both the DC and AC components of the signal under test can pass. ▹GND: Both the DC and AC components of the signal under test are blocked

Return	The query returns AC,DC,GND
Example	:CHANnel1:COUPling AC /*Select AC coupling mode*/ :CHANnel1:COUPling? /*Query returns AC*/

1.3 CHANnel<n>:DISPlay

Command	:CHANnel<n>:DISPlay <bool> :CHANnel<n>:DISPlay?
Description	Turn on or off the specified channel or query the switch status of the specified channel.

Parameters	:<bool> ::= {{1 | ON} | {0 | OFF}}
	:<n>::= {1 | 2 | 3 | 4}

Return	The query returns 0 or 1.
Example	:CHANnel1:DISPlay ON /*Turn on CH1*/ :CHANnel1:DISPlay? /*The query returns 1*/

1.4 CHANnel<n>:INVert

Command	:CHANnel<n>:INVert <bool> :CHANnel<n>:INVert?
Description	Turn the waveform inversion of the specified channel on or off or query the switching status of the waveform inversion of the specified channel.

Parameters	:<bool> ::= {{1 | ON} | {0 | OFF}}
	:<n>::= {1 | 2 | 3 | 4}

Return	The query returns 0 or 1.
Example	:CHANnel1:INVert ON /*Turn on the waveform inversion of CH1*/ :CHANnel1:INVert? /*The query returns 1*/

1.5 CHANnel<n>:OFFSet

Command	:CHANnel<n>:OFFSet <offset> [<suffix>] :CHANnel<n>:OFFSet?
Description	Sets or queries the vertical displacement of the specified channel. The default unit is V.

Parameters	:<offset> ::= Vertical offset value in NR3 format
	:<suffix> ::= {V | mV}
	:<n>::= {1 | 2 | 3 | 4}

State	▹The set vertical displacement value is affected by the vertical gear and probe ratio. ▹The range of legal values varies with the vertical scale and probe ratio. If 21 / 136 you set an offset outside the legal range, the offset value will be automatically set to the closest legal value.
Return	The query returns the vertical displacement value in scientific notation.
Example	:CHANnel1:OFFSet 1V /*Set the vertical offset of CH1 to 1V*/ :CHANnel1:OFFSet? /*The query returns 1*/

1.6 CHANnel<n>:SCALe

Command	:CHANnel<n>:SCALe <scale> [<suffix>] :CHANnel<n>:SCALe?
Description	Sets or queries the vertical scale of the specified channel. The default unit is V.

Parameters	:<scale> ::= vertical units per division in NR3 format
	:<suffix> ::= {V | mV}
	:<n>::= {1 | 2 | 3 | 4}

State	▷CHANnel<n>:SCALe command to setup each channel's vertical scale.The settable range of the vertical scale is related to the currently set probe ratio (set by the: ▷CHANnel : PROBe command).
Return	The query returns the vertical scale value in scientific notation.
Example	:CHANnel1:SCALe 1V /*Set the vertical scale to 1V*/ :CHANnel1:SCALe?/*The query returns 1.000e + 00*/

1.7 CHANnel<n>:PROBe

Command	:CHANnel<n>:PROBe <atten> :CHANnel<n>:PROBe?
Description	Set or query the probe factor of the specified channel.

Parameters	:<atten> ::= {1 | 10 | 100 | 1000}
	:<n>::= {1 | 2 | 3 | 4}

State	▷Set the probe factor, which is displayed after multiplying the acquired signal by a specified multiple (does not affect the actual amplitude of the signal). ▷Setting the probe ratio affects the settable range of the current vertical scale.
Return	The query returns the probe attenuation ratio in scientific notation.
Example	:CHANnel1:PROBe 10 /*Set the probe attenuation ratio to 10X*/ :CHANnel1:PROBe? /*The query returns 1.000000e + 01*/

1.8 CHANnel<n>:VERNier

Command	:CHANnel<n>:VERNier <bool> :CHANnel<n>:VERNier?
Description	Turn the fine adjustment function of the vertical scale of the specified channel on or off, or query the status of the fine adjustment function of the vertical scale of the specified channel.

Parameters	:<bool> ::= {{1 | ON} | {0 | OFF}}
	:<n>::= {1 | 2 | 3 | 4}

State	The trim setting is off by default. At this time, you can only set the vertical scale in 1-2-5 steps, that is, 500u, 1mV, 2mV, 5mV, 10mV ... 10V (probe ratio is 1X). When the trim setting is on, you can further adjust the vertical scale within a smaller range to improve vertical resolution. If the amplitude of the input waveform is slightly larger than the full scale in the current scale, and the amplitude displayed by the waveform of the next gear is slightly lower, you can use fine adjustment to improve the waveform display amplitude to facilitate observation of signal details.
Return	The query returns 1 or 0.
Example	:CHANnel1:VERNier ON /*Turn on the fine adjustment function of CH1 vertical

	scale*/ :CHANnel1:VERNier? /*The query returns 1*/

2. TIMebase Command Subsystem

Command List

- TIMebase:WINDow:ENABle
- TIMebase:WINDow:POSition
- TIMebase:WINDow:SCALe
- TIMebase:POSition
- TIMebase:SCALe
- TIMebase:RANGe
- TIMebase:MODE

2.1 TIMebase:WINDow:ENABle

Command	:TIMebase:WINDow:ENABle <bool> :TIMebase:WINDow:ENABle?
Description	Turn on or off the dual window function, or query the status of the dual window.
Parameters	:<bool> ::= {{1 | ON} | {0 | OFF}
State	The dual window mode can be used to zoom in a section of the waveform horizontally in order to view the waveform details.
Return	The query returns ON or Off.
Example	:TIMebase:WINDow:ENABle ON /*Turn on delayed scanning*/ :TIMebase:WINDow:ENABle? /*The query returns 1*/

2.2 TIMebase:WINDow:POSition

Command	:TIMebase:WINDow:POSition <pos value> :TIMebase:WINDow:POSition?
Description	Set or query horizontal position of the zoomed view of the main scan.
Parameters	:<pos value> ::= The value of the horizontal position (in seconds)
State	The main scan range and the horizontal position of the main scan determine the range of this command. The value of this command must keep the zoomed view window within the main scan range.
Return	Returns the position of the current horizontal window in the zoomed view in scientific notation.
Example	:TIMebase:WINDow:POSition 0.001 /*Set the horizontal position to 0.001s*/ :TIMebase:WINDow:POSition? /*The query returns1.000e-03*/

2.3 TIMebase:WINDow:SCALe

Command	:TIMebase:WINDow:SCALe <scale_value> :TIMebase:WINDow:SCALe?
Description	Set or query the horizontal scale of the sub-window (microseconds/div).
Parameters	:<sacle_value> ::= The current number of microseconds per grid of the subwindow.
State	The main scan scale determines the range of this command, the maximum value is half of the main scan scale.
Return	Return the time base gear of the sub-window in scientific notation.
Example	:TIMebase:WINDow:SCALe 0.0005 /*Set the delay time base to 500us*/ :TIMebase:WINDow:SCALe? /*The query returns 5.000e-4*/

2.4 TIMebase:POSition

Command	:TIMebase:POSition <pos value> :TIMebase:POSition?
Description	Set or query the main time base offset.
Parameters	:<pos value> ::= offset value (in second).
Return	Return the main time base offset value in scientific notation.
Example	:TIMebase:POSition 0.0002 /*Set main time base to 200us*/ :TIMebase:POSition? /*The query returns 2.000e-4*/

2.5 TIMebase:SCALe

Command	:TIMebase:SCALe <scale value> :TIMebase:SCALe?
Description	Set or query the horizontal scale of the main window (seconds/div).
Parameters	:<scale value> ::= The current number of seconds per grid in the main window
Return	Return to the main window time base gear in scientific notation.
Example	:TIMebase:SCALe 0.0005 /*Set the main time base to 500us*/ :TIMebase:SCALe? /*The query returns 5.000e-4*/

2.6 TIMebase:RANGe

Command	:TIMebase:RANGe <range value> :TIMebase:RANGe?
Description	Set or query the full-scale horizontal time of the main window.
Parameters	:<range value> ::= range value (in second).
Return	Return the full-scale time base range of the main window in scientific notation.
Example	:TIMebase:RANGe 0.0016 /*Set main time base to100us*/ :TIMebase:RANGe? /*The query returns 1.600e-03*/

2.7 TIMebase:MODE

Command	:TIMebase:MODE <value> :TIMebase:MODE?
Description	Sets or queries the horizontal time base mode.

Parameters	:<value> ::= <MAIN | XY | ROLL>
State	▷MAIN: YT mode ▷XY: XY mode ▷ROLL: ROLL mode
Return	The query returns MAIN, XY, or ROLL.
Example	:TIMebase:MODE XY /*Set the horizontal time base mode to XY mode*/ :TIMebase:MODE? /*The query returns XY*/

3. ACQuire Command Subsystem

Command List

- ACQuire:POINts
- ACQuire:TYPE
- ACQuire:SRATe?
- ACQuire:COUNt

3.1 ACQuire:POINts

Command	:ACQuire:POINts <value> :ACQuire:POINts?	
Description	Set or query the storage depth of the oscilloscope.	
Parameters	:<value> ::= memory depth value.	
State	value	Memory depth
	4000	4K
	40000	40K
	400000	400K
	4000000	4M
	8000000	8M
Return	The query returns the actual number of points (integer).	
Example	:ACQuire:POINts 4000 /*Set memory depth to 4k*/ :ACQuire:POINts? /*The query returns 4000*/	

3.2 ACQuire:TYPE

Command	:ACQuire:TYPE <value> :ACQuire:TYPE?
Description	Set or query the acquisition method of oscilloscope samples
Parameters	:<value> ::= < NORMal | AVERage | PEAK | HRESolution>
State	▷NORMal (normal): In this mode, the oscilloscope samples the signal at equal time intervals to reconstruct the waveform. For most waveforms, using this mode produces the best display results. ▷AVERages: In this mode, the oscilloscope averages waveforms that have been **18** / **136**

	sampled multiple times to reduce random noise on the input signal and improve vertical resolution. The number of averages can be set by the: ACQuire: AVERages command. The higher the number of averages, the lower the noise and the higher the vertical resolution, but the slower the response of the displayed waveform 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 signal envelope or narrow pulses that may be lost. Using this mode can avoid aliasing of the signal, but the display noise is relatively large. ▷HRESolution (high resolution): This mode uses an oversampling technique to average the neighboring points of the sampled waveform to reduce random noise on the input signal and produce a smoother waveform on the screen. It is usually used when the sampling rate of the digitizer is higher than the storage rate of the acquisition memory.
Return	The query returns NORM, AVERage, PEAK, or HRESolution.
Example	:ACQuire:TYPE AVERage /*Set the acquisition method to average*/ :ACQuire:TYPE? /*The query returns AVERage*/

3.3 ACQuire:SRATe?

Command	:ACQuire:SRATe?
Description	Query the current sampling rate.
State	The number of waveform points sampled per second.
Return	The query returns the sample rate as a real number.
Example	:ACQuire:SRATe? /*The query returns 1.25e+06*/

3.4 ACQuire:COUNt

Command	:ACQuire:COUNt <value> :ACQuire:COUNt?
Description	Set or query the average number of times in the average acquisition mode
Parameters	:<value> ::= <4|8|16|32|64|128>
Return	The query returns current average times.
Example	:ACQuire:COUNt 64 /*Set the average number to 64*/ :ACQuire:COUNt? /*The query returns 64*/

4. TRIGger Command Subsystem

Command List

- TRIGger:FORCe
- TRIGger:MODE
- TRIGger:STATus?
- TRIGger:SWEep
- TRIGger:HOLDoff
- TRIGger:EDGe

- TRIGger:PULSe
- TRIGger:SLOPe
- TRIGger:TV
- TRIGger:TIMeout
- TRIGger:WINDOw
- TRIGger:INTERVAl
- TRIGger:UNDER_Am
- TRIGger:UART
- TRIGger:CAN
- TRIGger:LIN
- TRIGger:IIC
- TRIGger:SPI
- TRIGger:PATTern

4.1 TRIGger:FORCe

Command	:TRIGger:FORCe
Description	Select or query the force trigger.
State	Even if the trigger conditions are not met, this command will let the oscilloscope acquire waveforms

4.2 TRIGger:MODE

Command	:TRIGger:MODE <mode> :TRIGger:MODE?
Description	Select or query the trigger type.
Parameters	:<mode> ::= < EDGE | PULSe | TV | SLOPe | TIMeout | WINdow | PATTern | INTerval | UNDerthrow | UART | LIN | CAN | SPI | IIC>
Return	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? /*The query returns SLOPe*/

4.3 TRIGger:STATus?

Command	:TRIGger:STATus?
Description	Query the current trigger status.
Return	Return TRIGed, NOTRIG

4.4 TRIGger:SWEep

Command	:TRIGger:SWEep <value> :TRIGger:SWEep?
Description	Set or query the trigger mode.
Parameters	:<value> ::= < AUTO | NORMal | SINGle>
State	▹AUTO: Automatic triggering, waveform display regardless of whether trigger conditions are met. ▹NORMal: Normal trigger, display the waveform when the trigger condition is met, keep the original waveform display when the trigger condition is not met, and wait for the next trigger. ▹SINGle: Single trigger. The oscilloscope waits for the trigger, displays the waveform when the trigger conditions are met, and then stops.
Return	The query returns AUTO, NORMal, or SINGle.
Example	:TRIGger:SWEep SINGle /*Select single trigger mode*/ :TRIGger:SWEep? /*The query returns SINGle*/

4.5 TRIGger:HOLDoff

Command	:TRIGger:HOLDoff <value> :TRIGger:HOLDoff?
Description	Set or query the holdoff time.
Parameters	:<value> ::= Trigger holdoff time (in second)
State	Trigger holdoff can stably trigger complex waveforms (such as pulse series). Holdoff time is the time the oscilloscope waits to re-enable the trigger circuit. The oscilloscope will not trigger until the holdoff time is over. When the trigger method is video trigger, timeout trigger, setup hold, UART, LIN, CAN, IIC or SPI, this setting is not available.
Return	The query returns the trigger holdoff time in scientific notation.
Example	:TRIGger:HOLDoff 0.0000002 /*Set trigger holdoff time to 200ns*/ :TRIGger:HOLDoff? /*The query returns 2.000000e-07*/

4.6 TRIGger:EDGe

Command List

- TRIGger:EDGe:SOURce
- TRIGger:EDGe:SLOPe
- TRIGger:EDGe:LEVel

4.6.1 TRIGger:EDGe:SOURce

Command	:TRIGger:EDGe:SOURce <source> :TRIGger:EDGe:SOURce?
Description	Set or query the trigger source for edge trigger.
Parameters	:<source> ::= < CHANnel1 | CHANnel2 | CHANnel3 | CHANnel4 | EXT/10>
Return	The query returns CHANnel1, CHANnel2, CHANnel3, CHANnel4, EXT/10
Example	:TRIGger:EDGe:SOURce CHANnel1 / * Set the trigger source to CH1 * / :TRIGger:EDGe:SOURce? / * The query returns CHANnel1 * /

4.6.2 TRIGger:EDGe:SLOPe

Command	:TRIGger:EDGe:SLOPe <slope> :TRIGger:EDGe:SLOPe?
Description	Set or query the type of edge trigger.
Parameters	:<slope> ::= < RISIng| FALLing | EITHer >
State	▹RISIng: rising edge ▹FALLing: falling edge ▹EITHer: rising or falling edge
Return	The query returns RISIng, FALLing, EITHer
Example	:TRIGger:EDGe:SLOPe RISIng /*Set the edge type to rising edge*/ :TRIGger:EDGe:SLOPe? /*The query returns RISIng*/

4.6.3 TRIGger:EDGe:LEVel

Command	:TRIGger:EDGe:LEVel <level> :TRIGger:EDGe:LEVel?
Description	Set or query the trigger level when the edge triggers.
Parameters	:<level> ::= trigger level (unit is V)
Return	The query returns the trigger level value in scientific notation.
Example	:TRIGger:EDGe:LEVel 0.16 /*Set the trigger level to 160mV*/ :TRIGger:EDGe:LEVel? /*The query returns 1.600e-1*/

4.7 TRIGger:PULSe

Command List

- TRIGger:PULSe:SOURce
- TRIGger:PULSe:POLarity
- TRIGger:PULSe:WHEN
- TRIGger:PULSe:WIDth
- TRIGger:PULSe:LEVel

4.7.1 TRIGger:PULSe:SOURce

Command	:TRIGger:PULSe:SOURce <source> :TRIGger:PULSe:SOURce?
Description	Set or query the trigger source of the pulse width trigger.
Parameters	:<source> ::= < CHANnel1 | CHANnel2 | CHANnel3 | CHANnel4>
Return	The query returns CHANnel1, CHANnel2, CHANnel3, CHANnel4
Example	:TRIGger:PULSe:SOURce CHANnel1 /*set trigger source to CH1*/ :TRIGger:PULSe:SOURce? /*The query returns CHANnel1*/

4.7.2 TRIGger:PULSe:POLarity

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

Description	Set or query the trigger polarity of pulse width trigger
Parameters	:<polarity> ::= < POSItive | NEGAtive>
State	▷POSItive: Positive pulse trigger ▷NEGAtive: negative pulse trigger
Return	The query returns POSItive, NEGAtive
Example	:TRIGger:PULSe:POLarity POSItive /*Set positive pulse trigger*/ :TRIGger:PULSe:POLarity? /*The query returns POSItive*/

4.7.3 TRIGger:PULSe:WHEN

Command	:TRIGger:PULSe:WHEN <when> :TRIGger:PULSe:WHEN?
Description	Set or query the trigger conditions for pulse width trigger.
Parameters	<when> ::= < EQUAl| NEQUal| GREAt| LESS>
State	▷EQUAl = (equal to time value): Only when the positive or negative pulse width of the input signal is equal to the set pulse width can it trigger. [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 the [pulse width error be 5%] be triggered. GREAt ＞(greater than the 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 the [pulse width error be 5%] be triggered.
Return	The query returns EQUAl, NEQUal, GRAt, LESS
Example	:TRIGger:PULSe:WHEN NEQUal /*Set trigger condition to NEQUal*/ :TRIGger:PULSe:WHEN? /*The query returns NEQUal*/

4.7.4 TRIGger:PULSe:WIDth

Command	:TRIGger:PULSe:WIDth <value> :TRIGger:PULSe:WIDth?
Description	Set or query the pulse width triggering time.
Parameters	:<value> ::=pulse width trigger (in second)
Return	The query returns the pulse width value in scientific notation
Example	:TRIGger:PULSe:WIDth 0.000003 /Set the pulse width value to 3μs*/ :TRIGger:PULSe:WIDth? /*The query returns 3.000000e-06*/

4.7.5 TRIGger:PULSe:LEVel

Command	:TRIGger:PULSe:LEVel <level> :TRIGger:PULSe:LEVel?
Description	Set or query the trigger level when pulse width trigger.
Parameters	:<level> ::=trigger level value (in V)
Return	The query returns the trigger level value in scientific notation.
Example	:TRIGger:PULSe:LEVel 0.16 /*Set the trigger level to 160mV*/ :TRIGger:PULSe:LEVel? /*The query returns 1.600000e-01*/

4.8 TRIGger:SLOPe

Command List

- TRIGger:SLOPe:SOURce
- TRIGger:SLOPe:POLarity
- TRIGger:SLOPe:WHEN
- TRIGger:SLOPe:WIDth
- TRIGger:SLOPe:ALEVel
- TRIGger:SLOPe:BLEVel

4.8.1 TRIGger:SLOPe:SOURce

Command	:TRIGger:SLOPe:SOURce <source> :TRIGger:SLOPe:SOURce?
Description	Set or query the trigger source of slope trigger
Parameters	:<source> ::= < CHANnel1 | CHANnel2 | CHANnel3 | CHANnel4>
Return	The query returns CHANnel1, CHANnel2, CHANnel3, CHANnel4
Example	:TRIGger:SLOPe:SOURce CHANnel1 /*Set trigger source to CH1*/ :TRIGger:SLOPe:SOURce? /*The query returns CHANnel1*/

4.8.2 TRIGger:SLOPe:POLarity

Command	:TRIGger:SLOPe:POLarity <polarity> :TRIGger:SLOPe:POLarity?
Description	Set or query the trigger polarity of the slope trigger.
Parameters	:<polarity> ::= < POSItive | NEGAtive>
State	POSItive: Positive slope trigger NEGAtive: negative slope trigger
Return	The query returns POSItive, NEGAtive
Example	:TRIGger:SLOPe:POLarity POSItive /*Set positive slope trigger*/ :TRIGger:SLOPe:POLarity? /*The query returns POSItive*/

4.8.3 TRIGger:SLOPe:WHEN

Command	:TRIGger:SLOPe:WHEN <when> :TRIGger:SLOPe:WHEN?
Description	Set or query the trigger condition of the slope trigger.
Parameters	:<when> ::= < EQUAl| NEQUal| GREAt| LESS>
State	▹EQUAl = (equal to time value): Only when the positive or negative pulse width of the input signal is equal to the set pulse width can it trigger. [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 the [pulse width error be 5%] be triggered. ▹GREAt＞(greater than the 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 the [pulse width error be 5%] be triggered.

Return	The query returns EQUAl, NEQUal, GRAt, LESS
Example	:TRIGger:SLOPe:WHEN NEQUal /*Set trigger condition to NEQUal*/ :TRIGger:SLOPe:WHEN? /*The query returns NEQUal*/

4.8.4 TRIGger:SLOPe:WIDth

Command	:TRIGger:SLOPe:WIDth <value> :TRIGger:SLOPe:WIDth?
Description	Set or query the trigger condition of the slope trigger.
Parameters	:<value> ::=trigger condition value (in second)
Return	The query returns the time value in scientific notation.
Example	:TRIGger:SLOPe:WIDth 0.000003 /*Set the width value to 3μs*/ :TRIGger:SLOPe:WIDth? /*The query returns 3.000000e-06*/

4.8.5 TRIGger:SLOPe:ALEVel

Command	:TRIGger:SLOPe:ALEVel <level> :TRIGger:SLOPe:ALEVel?
Description	Set or query the upper limit of the trigger level when the slope triggers.
Parameters	:<level> ::=trigger level upper limit value (in V)
Return	The query returns the upper limit value of the trigger level in scientific notation.
Example	:TRIGger:SLOPe:ALEVel 0.16 /*Set the trigger level to 160mV*/ :TRIGger:SLOPe:ALEVel? /*The query returns 1.600000e-01*/

4.8.6 TRIGger:SLOPe:BLEVel

Command	:TRIGger:SLOPe:BLEVel <level> :TRIGger:SLOPe:BLEVel?
Description	Set or query the lower limit of the trigger level when the slope triggers.
Parameters	:<level> ::=trigger level lower limit value (in V)
Return	The query returns the lower limit value of the trigger level in scientific notation.
Example	:TRIGger:SLOPe:BLEVel 0.16 /*Set the trigger level to lower limit*/ :TRIGger:SLOPe:BLEVel? /*The query returns 1.600000e-01*/

4.9 TRIGger:TV

Command List

- TRIGger:TV:SOURce
- TRIGger:TV:POLarity
- TRIGger:TV:MODE
- TRIGger:TV:LINE
- TRIGger:TV:STANdard
- TRIGger:VIDeo:LEVel

4.9.1 TRIGger:TV:SOURce

Command	:TRIGger:TV:SOURce <source> :TRIGger:TV:SOURce?
Description	Select or query the trigger source of the video trigger.
Parameters	:<source> ::= < CHANnel1 | CHANnel2 | CHANnel3 | CHANnel4>
Return	The query returns CHANnel1, CHANnel2, CHANnel3, CHANnel4
Example	:TRIGger:TV:SOURce CHANnel1 /*Settrigger source to CH1*/ :TRIGger:TV:SOURce? /*The query returns CHANnel1*/

4.9.2 TRIGger:TV:POLarity

Command	:TRIGger:TV:POLarity <polarity> :TRIGger:TV:POLarity?
Description	Select or query the video polarity when the video is triggered.
Parameters	:<polarity> ::= < POSItive | NEGAtive>
State	POSItive: Positive video trigger NEGAtive: negative video trigger
Return	The query returns POSItive, NEGAtive.
Example	:TRIGger:TV:POLarity POSItive /*Set the video polarity to positive polarity*/ :TRIGger:TV:POLarity? /*Query returns POSItive*/

4.9.3 TRIGger:TV:MODE

Command	:TRIGger:TV:MODE <mode> :TRIGger:TV:MODE?
Description	Set or query the synchronization type when the video is triggered.
Parameters	:<mode> ::= < ALINes|LINEs|FIEld1|FIEld2|AFIelds>
State	▹ALINes: trigger on all horizontal sync pulses. ▹LINEs: For NTSC and PAL / SECAM video standards, trigger on a specified line in the odd or even field. ▹FIDld1: Trigger on the rising edge of the first sawtooth wave in the odd field. ▹FIDld2: Trigger on the first rising edge of the sawtooth wave in the even field.
Return	The query returns ALINes, LINEs, FIEld1, FIEld2, AFIelds
Example	:TRIGger:TV:MODE FIEld1 /*Set the synchronization type to an odd field*/ :TRIGger:TV:MODE? /*The query returns FIEld1*/

4.9.4 TRIGger:TV:LINE

Command	:TRIGger:TV:LINE <line> :TRIGger:TV:LINE?
Description	Sets or queries the line number when the synchronization type is the specified line when the video is triggered.
Parameters	:<line> ::=the line number of the specified line
State	▹NTSC:1-525 ▹PAL/SECAM:1-625
Return	The query returns an integer.
Example	:TRIGger:TV:LINE 100 /*Set the line number to 100*/ :TRIGger:TV:LINE? /*The query returns 100*/

4.9.5 TRIGger:TV:STANdard

Command	:TRIGger:TV:STANdard <standard> :TRIGger:TV:STANdard?
Description	Set or query the video standard of the video
Parameters	:<standard> ::= < NTSC | PAL >
Return	The query returns NTSC, PAL
Example	:TRIGger:TV:STANdard NTSC /*Select NTSC video standard*/ :TRIGger:TV:STANdard? /*The query returns NTSC*/

4.9.6 TRIGger:VIDeo:LEVel

Command	:TRIGger:VIDeo:LEVel <level> :TRIGger:VIDeo:LEVel?
Description	Set orquery the trigger level when the video is triggered.
Parameters	:<level> ::= trigger level value (in V)
Return	The query returns the trigger level in scientific notation.
Example	:TRIGger:VIDeo:LEVel 0.16 /*Set the trigger level to 160mV*/ :TRIGger:VIDeo:LEVel? /*The query returns 1.600000e-01*/

4.10 TRIGger:TIMeout

Command List

- TRIGger:TIMeout:SOURce
- TRIGger:TIMeout:LEVel
- TRIGger:TIMeout:WIDth
- TRIGger:TIMeout:POLarity

4.10.1 TRIGger:TIMeout:SOURce

Command	:TRIGger:TIMeout:SOURce <source> :TRIGger:TIMeout:SOURce?
Description	Sets or queries the trigger source of the timeout trigger.
Parameters	:<source> ::= < CHANnel1 | CHANnel2 | CHANnel3 | CHANnel4>
Return	The query returns CHANnel1, CHANnel2, CHANnel3, CHANnel4
Example	:TRIGger:TIMeout:SOURce CHANnel1 /*Set trigger source to CH1*/ :TRIGger:TIMeout:SOURce? /*The query returns CHANnel1*/

4.10.2 TRIGger:TIMeout:LEVel

Command	:TRIGger:TIMeout:LEVel <level> :TRIGger:TIMeout:LEVel?
Description	Set or query the trigger level when the timeout triggers.
Parameters	:<level> ::= trigger level value (in V)
Return	Returns the trigger level value in scientific notation
Example	:TRIGger:TIMeout:LEVel 0.16 /*Set trigger level 160mV*/ :TRIGger:TIMeout:LEVel? /*The query returns 1.600000e-01*/

4.10.3 TRIGger:TIMeout:WIDth

Command	:TRIGger:TIMeout:WIDth <value> :TRIGger:TIMeout:WIDth?
Description	Sets or queries the timeout period triggered by the timeout.
Parameters	:<value> ::=trigger timeout value (range: 8ns-10s)
Return	The query returns the timeout value in scientific notation.
Example	:TRIGger:TIMeout:WIDth 0.000003 /*Set the timeout time to 3μs*/ :TRIGger:TIMeout:WIDth? /*The query returns 3.000000e-06*/

4.10.4 TRIGger:TIMeout:POLarity

Command	:TRIGger:TIMeout:POLarity <polarity> :TRIGger:TIMeout:POLarity?
Description	Sets or queries the polarity of edge triggered by the timeout.
Parameters	:<polarity> ::= < POSItive | NEGAtive>
State	▹POSItive: Start the timing by the trigger level on the rising edge of the input signal. ▹NEGAtive: Starts timing with the trigger level on the falling edge of the input signal.
Return	The query returns POSItive, NEGAtive.
Example	:TRIGger:TIMeout:POLarity POSItive /*Set to positive polarity of the trigger*/ :TRIGger:TIMeout:POLarity? /*The query returns POSItive*/

4.11 TRIGger:WINDOw

Command List

- TRIGger:WINDOw:SOURce
- TRIGger:WINDOw:ALEVel
- TRIGger:WINDOw:BLEVel

4.11.1 TRIGger:WINDOw:SOURce

Command	:TRIGger:WINDOw:SOURce <source> :TRIGger:WINDOw:SOURce?
Description	Set or query the trigger source triggered by the window.
Parameters	:<source> ::= < CHANnel1 | CHANnel2 | CHANnel3 | CHANnel4>
Return	The query returns CHANnel1, CHANnel2, CHANnel3, CHANnel4.
Example	:TRIGger:WINDOw:SOURce CHANnel1 /*Set trigger source to CH1*/ :TRIGger:WINDOw:SOURce? /*The query returns CHANnel1*/

4.11.2 TRIGger:WINDOw:ALEVel

Command	:TRIGger:WINDOw:ALEVel <level> :TRIGger:WINDOw:ALEVel?
Description	Set or query the upper limit of the trigger level when the super window is sent.
Parameters	:<level> ::=trigger level upper limit value (in V)
Return	The query returns the upper trigger level in scientific notation.

Example	:TRIGger:WINDOw:ALEVel 0.16 /*Set the trigger level upper limit to 160mV*/ :TRIGger:WINDOw:ALEVel? /*The query returns 1.600000e-01*/

4.11.3 TRIGger:WINDOw:BLEVel

Command	:TRIGger:WINDOw:BLEVel <level> :TRIGger:WINDOw:BLEVel?
Description	Set or query the lower limit of the trigger level when the amplitude trigger is triggered.
Parameters	:<level> ::=trigger level lower limit value (in V)
Return	The query returns the lower trigger level in scientific notation.
Example	:TRIGger:WINDOw:BLEVel 0.16 /*Set the trigger level lower limit to 160mV*/ :TRIGger:WINDOw:BLEVel? /*The query returns 1.600000e-01*/

4.12 TRIGger:INTERVAl

Command List

- TRIGger:INTERVAl:SOURce
- TRIGger:INTERVAl:SLOp
- TRIGger:INTERVAl:WHEN
- TRIGger:INTERVAl:TIME
- TRIGger:INTERVAl:ALEVel

4.12.1 TRIGger:INTERVAl:SOURce

Command	:TRIGger:INTERVAl:SOURce <source> :TRIGger:INTERVAl:SOURce?
Description	Set or query the trigger source for interval trigger
Parameters	:<source> ::= < CHANnel1 | CHANnel2 | CHANnel3 | CHANnel4>
State	Triggered when the interval between two consecutive rising edges (or falling edges) meets the set time condition (<,>, =,! =)
Return	The query returns CHANnel1, CHANnel2, CHANnel3, CHANnel4
Example	:TRIGger:INTERVAl:SOURce CHANnel1 /*Set the trigger source to CH1*/ :TRIGger:INTERVAl:SOURce? /*The query returns CHANnel1*/

4.12.2 TRIGger:INTERVAl:SLOp

Command	:TRIGger:INTERVAl:SLOp <slope> :TRIGger:INTERVAl:SLOp?
Description	Set or query the type of edge triggered by the interval
Parameters	:<slope> ::= < RISIng| FALLing>
State	▹RISIng: rising edge trigger ▹FALLing: falling edge trigger ▹DOUBle: Double edge trigger
Return	The query returns RISIng, FALLing, DOUBle
Example	:TRIGger:INTERVAl:SLOp RISIng /Set the edge type to rising edge*/ :TRIGger:INTERVAl:SLOp? /*The query returns POSItive*/

4.12.3 TRIGger:INTERVAl:WHEN

Command	:TRIGger:INTERVAl:WHEN <when> :TRIGger:INTERVAl:WHEN?
Description	Set or query the trigger condition of the interval trigger
Parameters	:<when> ::= < EQUAl| NEQUal| GREAt| LESS>
State	The interval between two consecutive rising (or falling) edges meets the set time condition(<,>, =,! =)
Return	The query returns EQUAl, NEQUal, GRAt, LESS
Example	:TRIGger:INTERVAl:WHEN NEQUal /*Set the trigger condition to NEQUal*/ :TRIGger:INTERVAl:WHEN? /*The query returns NEQUal*/

4.12.4 TRIGger:INTERVAl:TIME

Command	:TRIGger:INTERVAl:TIME <value> :TRIGger:INTERVAl:TIME?
Description	Set or query the time value when the interval is triggered
Parameters	:<value> ::=trigger time value (in second, 8ns-10s)
Return	Returns the time value in scientific notation
Example	:TRIGger:INTERVAl:TIME 0.000003 /*Set the pulse width value 3us*/ :TRIGger:INTERVAl:TIME? /*The query returns 3.000000e-06*/

4.12.5 TRIGger:INTERVAl:ALEVel

Command	:TRIGger:INTERVAl:ALEVel <level> :TRIGger:INTERVAl:ALEVel?
Description	Set or query the trigger level for interval trigger
Parameters	:<level> ::=trigger level value (in V)
Return	Returns the trigger level value in scientific notation
Example	:TRIGger:INTERVAl:ALEVel 0.16 /*Set trigger level 160mV*/ :TRIGger:INTERVAl:ALEVel? /*The query returns 1.600000e-01*/

4.13 TRIGger:UNDER_Am

Runt trigger is used to trigger a pulse that has crossed one trigger level but not another trigger level.

Command List

- TRIGger:UNDER_Am:SOURce
- TRIGger:UNDER_Am:POLarity
- TRIGger:UNDER_Am:WHEN
- TRIGger:UNDER_Am:TIME
- TRIGger:UNDER_Am:ALEVel
- TRIGger:UNDER_Am:BLEVel

4.13.1 TRIGger:UNDER_Am:SOURce

Command	:TRIGger:UNDER_Am:SOURce <source> :TRIGger:UNDER_Am:SOURce?
Description	Set or query the trigger source of runt trigger.
Parameters	:<source> ::= < CHANnel1 | CHANnel2 | CHANnel3 | CHANnel4>
Return	The query returns CHANnel1, CHANnel2, CHANnel3, CHANnel4.
Example	:TRIGger:UNDER_Am:SOURce CHANnel1 /*Set the trigger source to CH1*/ :TRIGger:UNDER_Am:SOURce? /*The query returns CHANnel1*/

4.13.2 TRIGger:UNDER_Am:POLarity

Command	:TRIGger:UNDER_Am:POLarity <polarity> :TRIGger:UNDER_Am:POLarity?
Description	Set or query the pulse polarity of runt trigger.
Parameters	:<polarity> ::= < POSItive | NEGAtive>
State	▹POSitive: trigger on a positive runt pulse. ▹NEGative: trigger on negative runt pulses.
Return	The query returns POSItive or NEGAtive.
Example	:TRIGger:UNDER_Am:POLarity POSItive /*Set the pulse polarity to negative polarity*/ :TRIGger:UNDER_Am:POLarity? /*The query returns POSItive*/

4.13.3 TRIGger:UNDER_Am:WHEN

Command	:TRIGger:UNDER_Am:WHEN <when> :TRIGger:UNDER_Am:WHEN?
Description	Sets or queries the qualifier for runt trigger.
Parameters	:<when> ::= < EQUAl| NEQUal| GREAt| LESS>
Return	The query returns EQUAl, NEQUal, GRAt, LESS
Example	:TRIGger:UNDER_Am:WHEN NEQUal /*Set the qualifier to NEQUal*/ :TRIGger:UNDER_Am:WHEN? /*The query returns NEQUal*/

4.13.4 TRIGger:UNDER_Am:TIME

Command	:TRIGger:UNDER_Am:TIME <value> :TRIGger:UNDER_Am:TIME?
Description	Set or query the trigger time.
Parameters	:<value> ::=trigger time value (in second, range: 8ns-10s)
Return	Returns the trigger time value in scientific notation
Example	:TRIGger:UNDER_Am:TIME 0.000003 /*Set the time value to 3us*/ :TRIGger:UNDER_Am:TIME? /*The query returns 3.000000e-06*/

4.13.5 TRIGger:UNDER_Am:ALEVel

Command	:TRIGger:UNDER_Am:ALEVel <level> :TRIGger:UNDER_Am:ALEVel?
Description	Set or query the upper limit of the trigger level when the runt pulse is triggered.
Parameters	:<level> ::=trigger level upper limit value (in V)

Return	The query returns the upper trigger level in scientific notation.
Example	:TRIGger:UNDER_Am:ALEVel 0.16 /*Set the trigger level upper limit to 160mV*/ :TRIGger:UNDER_Am:ALEVel? /*The query returns 1.600000e-01*/

4.13.6 TRIGger:UNDER_Am:BLEVel

Command	:TRIGger:UNDER_Am:BLEVel <level> :TRIGger:UNDER_Am:BLEVel?
Description	Set or query the lower limit of the trigger level when the runt pulse is triggered.
Parameters	:<level> ::=trigger level lower limit value (in V)
Return	The query returns the lower trigger level in scientific notation.
Example	:TRIGger:UNDER_Am:BLEVel 0.16 /*Set the trigger level lower limit to 160mV*/ :TRIGger:UNDER_Am:BLEVel? /*The query returns 1.600000e-01*/

4.14 TRIGger:UART

Command List

- TRIGger:UART:SOURce
- TRIGger:UART:CONdition
- TRIGger:UART:BAUd
- TRIGger:UART:ALEVel
- TRIGger:UART:DATA
- TRIGger:UART:WIDTh
- TRIGger:UART:PARIty

4.14.1 TRIGger:UART:SOURce

Command	:TRIGger:UART:SOURce <source> :TRIGger:UART:SOURce?
Description	Set or query the trigger conditions for UART trigger.
Parameters	:<source> ::= < CHANnel1 | CHANnel2 | CHANnel3 | CHANnel4>
Return	The query returns CHANnel1, CHANnel2, CHANnel3, CHANnel4
Example	:TRIGger:UART:SOURce CHANnel1 /*Set the trigger source to CH1*/ :TRIGger:UART:SOURce? /*The query returns CHANnel1*/

4.14.2 TRIGger:UART:CONdition

Command	:TRIGger:UART:CONdition <condition> :TRIGger:UART:CONdition?
Description	Set or query the trigger conditions for UART trigger.
Parameters	:<condition> ::= < START| STOP| READ_DATA| PARITY_ERR| COM_ERR>
Return	The query returns START | STOP | READ_DATA | PARITY_ERR | COM_ERR
Example	:TRIGger:UART:CONdition START /*Set the trigger condition to START*/ :TRIGger:UART:CONdition? /*The query returns START*/

4.14.3 TRIGger:UART:BAUd

Command	:TRIGger:UART:BAUd <baud>

	:TRIGger:UART:BAUd?
Description	Set or query the baud rate triggered by the UART. The default unit is bps.
Parameters	:<baud> ::=<110|300|600|1200|2400|4800|9600|14400|19200|38400|57600|115200|230400|380400|460400|921600|USER>
Return	The query returns an integer or USER.
Example	:TRIGger:UART:BAUd 4800 /*Set the baud rate to 4800*/ :TRIGger:UART:BAUd? /*he query returns 4800*/

4.14.4 TRIGger:UART:ALEVel

Command	:TRIGger:UART:ALEVel <level> :TRIGger:UART:ALEVel?
Description	Set or query the trigger level when the UART triggers.
Parameters	:<level> ::= trigger level value (in V)
Return	The query returns the trigger level in scientific notation.
Example	:TRIGger:UART:ALEVel 0.16 /*Set trigger level 160mV*/ :TRIGger:UART:ALEVel? /*The query returns 1.600000e-01*/

4.14.5 TRIGger:UART:DATA

Command	:TRIGger:UART:DATA <data> :TRIGger:UART:DATA?
Description	Set or query the data value when the UART trigger condition is data.
Parameters	:<data> ::= (0-- (2^{n-1}-1))
State	n is the current data width, and the value range is 5, 6, 7, or 8.
Return	The query returns an integer.
Example	:TRIGger:UART:DATA 10 /*Set the data value to 10*/ :TRIGger:UART:DATA? /*The query returns 10*/

4.14.6 TRIGger:UART:WIDTh

Command	:TRIGger:UART:WIDTh <value> :TRIGger:UART:WIDTh?
Description	Sets or queries the data bit width when the UART trigger condition is data.
Parameters	:<value> ::= <5,6,7,8>
Return	The query returns 5, 6, 7, or 8.
Example	:TRIGger:UART:WIDTh 5 /*Set the data bit width to 5*/ :TRIGger:UART:WIDTh? /*The query returns 5*/

4.14.7 TRIGger:UART:PARIty

Command	:TRIGger:UART:PARIty <parity> :TRIGger:UART:PARIty?
Description	Set or query the verification mode when the UART trigger condition is an error frame or a verification error.
Parameters	:<parity> ::= < NONE| ODD| EVEN>
Return	The query returns EVEN, ODD, or NONE.
Example	:TRIGger:UART:PARIty EVEN /*Set the parity method to even parity*/ :TRIGger:UART:PARIty? /*The query returns EVEN*/

4.15 TRIGger:CAN

Command List

- TRIGger:CAN:SOURce
- TRIGger:CAN:IDLe
- TRIGger:CAN:BAUd
- TRIGger:CAN:CONdition
- TRIGger:CAN:ID
- TRIGger:CAN:DLC
- TRIGger:CAN:DATA
- TRIGger:CAN:ALEVel

4.15.1 TRIGger:CAN:SOURce

Command	:TRIGger:CAN:SOURce <source> :TRIGger:CAN:SOURce?
Description	Set or query the trigger source of CAN trigger.
Parameters	:<source> ::= < CHANnel1 | CHANnel2 | CHANnel3 | CHANnel4>
Return	The query returns CHANnel1, CHANnel2, CHANnel3, CHANnel4
Example	:TRIGger:CAN:SOURce CHANnel1 /*Set the trigger source to CH1*/ :TRIGger:CAN:SOURce? /*The query returns CHANnel1*/

4.15.2 TRIGger:CAN:IDLe

Command	:TRIGger:CAN:IDLe <idle> :TRIGger:CAN:IDLe?
Description	Set or query the idle level triggered by CAN
Parameters	:<idle> ::= <LOW | HIGH>
Return	The query returns LOW, HIGH
Example	:TRIGger:CAN:IDLe LOW /*Set idle low level*/ :TRIGger:CAN:IDLe? /*The query returns LOW*/

4.15.3 TRIGger:CAN:BAUd

Command	:TRIGger:CAN:BAUd <baud> :TRIGger:CAN:BAUd?
Description	Set or query the baud rate triggered by CAN. The default unit is bps.
Parameters	:<baud> ::=<10000|20000|33300|50000|62500|83300|100000|125000|250000|500000|800000|1000000|USER>
Return	The query returns an integer
Example	:TRIGger:CAN:BAUd 4800 /*Set the baud rate to 4800*/ :TRIGger:CAN:BAUd? /*The query returns 4800*/

4.15.4 TRIGger:CAN:CONdition

Command	:TRIGger:CAN:CONdition <condition> :TRIGger:CAN:CONdition?
Description	Set or query the trigger conditions for CAN trigger
Parameters	:<condition> ::= < FRAM_STARE | FRAM_REMO_ID | FRAM_DATA_ID | REMO/DATA_ID | DATA_ID/DATA | FRAM_REE | FRAM_OVERLOAD | ERR_ALL | ACK_ERR >
Return	Query returns FRAM_STARE | FRAM_REMO_ID | FRAM_DATA_ID |REMO/DATA_ID | DATA_ID/DATA | FRAM_REMO_ID_EXT |FRAM_DATA_ID_EXT | REMO/DATA_ID_EXT |DATA_ID/DATA_EXT | FRAM_REE | FRAM_OVERLOAD | ERR_ALL | ACK_ERR
Example	:TRIGger:CAN:CONdition FRAM_STARE /*Set FRAM_STARE*/ :TRIGger:CAN:CONdition? /*The query returns FRAM_STARE*/

4.15.5 TRIGger:CAN:ID

Command	:TRIGger:CAN:ID <id> :TRIGger:CAN:ID?
Description	Set or query the IDENTIFIER triggered by CAN
Parameters	:<id> ::= 0 -- 28
Return	The query returns an integer
Example	:TRIGger:CAN:ID 25 /*Set IDENTIFIER to 25*/ :TRIGger:CAN:ID? /*The query returns 25*/

4.15.6 TRIGger:CAN:DLC

Command	:TRIGger:CAN:DLC <dlc> :TRIGger:CAN:DLC?
Description	Set or query the data length code of CAN trigger
Parameters	:<dlc> ::= 4 digits
Return	The query returns an integer
Example	:TRIGger:CAN:DLC 10 /*Set the data length code to 10*/ :TRIGger:CAN:DLC? /*The query returns 10*/

4.15.7 TRIGger:CAN:DATA

Command	:TRIGger:CAN:DATA <index> <data> :TRIGger:CAN:DATA? <index>
Description	Set or query CAN trigger data value
Parameters	:<data> ::= 8 digits :<index> ::= data index 0-3
Return	The query returns an integer
Example	:TRIGger:CAN:DATA 2,10 /*Set the data with index 2 to 10*/ :TRIGger:CAN:DATA? 2 /*The query returns 10*/

4.15.8 TRIGger:CAN:ALEVel

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

Description	Set or query the trigger level during CAN trigger
Parameters	:<level> ::= trigger level value (in V)
Return	Returns the trigger level value in scientific notation
Example	:TRIGger:CAN:ALEVel 0.16 /*Set trigger level 160mV*/ :TRIGger:CAN:ALEVel? TRIGger:CAN:ALEVel?

4.16 TRIGger:LIN

Command List

- TRIGger:LIN:SOURce
- TRIGger:LIN:IDLe
- TRIGger:LIN:BAUd
- TRIGger:LIN:CONdition
- TRIGger:LIN:ID
- TRIGger:LIN:DATA
- TRIGger:LIN:ALEVel

4.16.1 TRIGger:LIN:SOURce

Command	:TRIGger:LIN:SOURce <source> :TRIGger:LIN:SOURce?
Description	Set or query the trigger source of the LIN trigger.
Parameters	:<source> ::= < CHANnel1 | CHANnel2 | CHANnel3 | CHANnel4>
Return	The query returns CHANnel1, CHANnel2, CHANnel3, CHANnel4
Example	:TRIGger:LIN:SOURce CHANnel1 /*Set the trigger source to CH1*/ :TRIGger:LIN:SOURce? /*The query returns CHANnel1*/

4.16.2 TRIGger:LIN:IDLe

Command	:TRIGger:LIN:IDLe <idle> :TRIGger:LIN:IDLe?
Description	Set or query the idle level triggered by LIN
Parameters	:<idle> ::= <LOW | HIGH>
Return	The query returns LOW, HIGH
Example	:TRIGger:LIN:IDLe LOW /*Set idle low level*/ :TRIGger:LIN:IDLe? /*The query returns LOW*/

4.16.3 TRIGger:LIN:BAUd

Command	:TRIGger:LIN:BAUd <baud> :TRIGger:LIN:BAUd?
Description	Set or query the baud rate triggered by LIN. The default unit is bps.
Parameters	:<baud> ::=<110|300|600|1200|2400|4800|9600|14400|19200|38400|57600|115200|230

	400|380400|460400|921600|USER>
Return	The query returns an integer
Example	:TRIGger:LIN:BAUd 4800 /*set the baud rate is 4800*/ :TRIGger:LIN:BAUd? /*The query returns 4800*/

4.16.4 TRIGger:LIN:CONdition

Command	:TRIGger:LIN:CONdition <condition> :TRIGger:LIN:CONdition?
Description	LIN trigger conditions set trigger or queries
Parameters	:<condition> ::= < INTERVAL_FIELD|SYNC_FIELD|ID_FIELD|DATA|IDENTIFIER|ID_DATA >
Return	The query returns INTERVAL_FIELD | SYNC_FIELD | ID_FIELD | DATA | IDENTIFIER | ID_DATA
Example	:TRIGger:LIN:CONdition DATA /*set INTERVAL_END*/ :TRIGger:LIN:CONdition? /*The query returns INTERVAL_END*/

4.16.5 TRIGger:LIN:ID

Command	:TRIGger:LIN:ID <id> :TRIGger:LIN:ID?
Description	Set or query LIN triggered identifier
Parameters	:<id> ::= 6 digits
Return	The query returns an integer
Example	:TRIGger:LIN:ID 25 /*Set identifier to 25*/ :TRIGger:LIN:ID? /*The query returns 25*/

4.16.6 TRIGger:LIN:DATA

Command	:TRIGger:LIN:DATA <index> <data> :TRIGger:LIN:DATA? <index>
Description	Set or query data values triggered by LIN.
Parameters	:<data> ::= 8 digits :<index> ::= data index 0-3
Return	The query returns an integer
Example	:TRIGger:LIN:DATA 2 ,10 /*Set the data with index 2 to 10*/ :TRIGger:LIN:DATA? 2 /*The query returns 10*/

4.16.7 TRIGger:LIN:ALEVel

Command	:TRIGger:LIN:ALEVel <level> :TRIGger:LIN:ALEVel?
Description	Set or query the trigger level during LIN trigger
Parameters	:<level> ::= trigger level value (in V)
Return	Returns the trigger level value in scientific notation
Example	:TRIGger:LIN:ALEVel 0.16 /*Set trigger level 160mV*/ :TRIGger:LIN:ALEVel? /*The query returns 1.600000e-01*/

4.17 TRIGger:IIC

Command List

- TRIGger:IIC:SDA:SOURce
- TRIGger:IIC:SCL:SOURce
- TRIGger:IIC:CONdition
- TRIGger:IIC:ADDer
- TRIGger:IIC:DATA
- TRIGger:IIC:ALEVel
- TRIGger:IIC:BLEVel

4.17.1 TRIGger:IIC:SDA:SOURce

Command	:TRIGger:IIC:SDA:SOURce <source> :TRIGger:IIC:SDA:SOURce?
Description	Set or query the channel source of the I2C triggered data line.
Parameters	:<source> ::= < CHANnel1 | CHANnel2 | CHANnel3 | CHANnel4>
Return	The query returns CHANnel1, CHANnel2, CHANnel3, CHANnel4
Example	:TRIGger:IIC:SDA:SOURce CHANnel1 /*Set the data source to CH1*/ :TRIGger:IIC:SDA:SOURce? /*The query returns CHANnel1*/

4.17.2 TRIGger:IIC:SCL:SOURce

Command	:TRIGger:IIC:SCL:SOURce <source> :TRIGger:IIC:SCL:SOURce?
Description	Set or query the channel source of the clock line triggered by I2C.
Parameters	:<source> ::= < CHANnel1 | CHANnel2 | CHANnel3 | CHANnel4>
Return	The query returns CHANnel1, CHANnel2, CHANnel3, CHANnel4
Example	:TRIGger:IIC:SCL:SOURce CHANnel1 /*Set the clock source to CH1*/ :TRIGger:IIC:SCL:SOURce? /*The query returns CHANnel1*/

4.17.3 TRIGger:IIC:CONdition

Command	:TRIGger:IIC:CONdition <condition> :TRIGger:IIC:CONdition?
Description	Set or query the trigger condition of IIC trigger.
Parameters	:<condition> ::= < START| STOP| ACK_LOST| ADDR_NO_ACK| RESTART| READ_DATA>
Return	The query returns START| STOP| ACK_LOST| ADDR_NO_ACK| RESTART| READ_DATA
Example	:TRIGger:IIC:CONdition START /*Set START */ :TRIGger:IIC:CONdition? /*The query returns START*/

4.17.4 TRIGger:IIC:ADDer

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

Description	Set or query the address value when the IIC trigger condition is an address or address data.
Parameters	:<addr> ::= 8 digits
Return	The query returns an integer.
Example	:TRIGger:IIC:ADDer 20 /*Set the address value to 20*/ :TRIGger:IIC:ADDer? /*The query returns START 20*/

4.17.5 TRIGger:IIC:DATA

Command	:TRIGger:IIC:DATA <index> <data> :TRIGger:IIC:DATA? <index>
Description	Set or query the data value when the IIC trigger condition is data or address data.
Parameters	:<data> ::= 8 digits :<index> ::= data index 0-8
Return	The query returns an integer.
Example	:TRIGger:IIC:DATA 2,10 /* Set the data with index 2 to 10*/ :TRIGger:IIC:DATA? 2 /* The query returns 10*/

4.17.6 TRIGger:IIC:ALEVel

Command	:TRIGger:IIC:ALEVel <level> :TRIGger:IIC:ALEVel?
Description	Set or query the trigger level of the clock line when I2C triggers.
Parameters	:<level> ::= trigger level value (in V)
Return	The query returns the trigger level in scientific notation.
Example	:TRIGger:IIC:ALEVel 0.16 /* Set the trigger level to 160mV */ :TRIGger:IIC:ALEVel? /* The query returns 1.600000e-01*/

4.17.7 TRIGger:IIC:BLEVel

Command	:TRIGger:IIC:BLEVel <level> :TRIGger:IIC:BLEVel?
Description	Set or query the trigger level of the data line when I2C triggers.
Parameters	:<level> ::= trigger level value (in V)
Return	The query returns the trigger level in scientific notation.
Example	:TRIGger:IIC:BLEVel 0.16 /* Set the trigger level to 160mV */ :TRIGger:IIC:BLEVel? /* The query returns 1.600000e-01*/

4.18 TRIGger:SPI

Command List

- TRIGger:SPI:SDA:SOURce
- TRIGger:SPI:SCL:SOURce
- TRIGger:SPI:SCK
- TRIGger:SPI:WIDth
- TRIGger:SPI:DATA

- TRIGger:SPI:MASK
- TRIGger:SPI:ALEVel
- TRIGger:SPI:BLEVel

4.18.1 TRIGger:SPI:SDA:SOURce

Command	:TRIGger:SPI:SDA:SOURce <source> :TRIGger:SPI:SDA:SOURce?
Description	Set or query the channel source of the data line triggered by SPI.
Parameters	:<source> ::= < CHANnel1 | CHANnel2 | CHANnel3 | CHANnel4>
Return	The query returns CHANnel1, CHANnel2, CHANnel3, CHANnel4
Example	:TRIGger:SPI:SDA:SOURce CHANnel1 /*Set the channel source of the data line to CH1*/ :TRIGger:SPI:SDA:SOURce? /*The query returns CHANnel1*/

4.18.2 TRIGger:SPI:SCL:SOURce

Command	:TRIGger:SPI:SCL:SOURce <source> :TRIGger:SPI:SCL:SOURce?
Description	Set or query the channel source of the clock line triggered by SPI.
Parameters	:<source> ::= < CHANnel1 | CHANnel2 | CHANnel3 | CHANnel4>
Return	The query returns CHANnel1, CHANnel2, CHANnel3, CHANnel4
Example	:TRIGger:SPI:SCL:SOURce CHANnel1 /*Set the channel source of the clock line to CH1*/ :TRIGger:SPI:SCL:SOURce? /*The query returns CHANnel1*/

4.18.3 TRIGger:SPI:SCK

Command	:TRIGger:SPI:SCK <slope> :TRIGger:SPI:SCK?
Description	Set or query the type of clock edge triggered by SPI.
Parameters	:<slope> ::= < Rising| Falling>
Return	The query returns Rising, Falling.
Example	:TRIGger:SPI:SCK Falling /*Set the clock edge to the falling edge*/ :TRIGger:SPI:SCK? /*The query returns Falling*/

4.18.4 TRIGger:SPI:WIDth

Command	:TRIGger:SPI:WIDth <width> :TRIGger:SPI:WIDth?
Description	Set or query the data bit width of the data channel under SPI trigger
Parameters	:<width> ::= 4--32
Return	The query returns an integer.
Example	:TRIGger:SPI:WIDth 20 /*Set the width to 20*/ :TRIGger:SPI:WIDth? /*The query returns 20*/

4.18.5 TRIGger:SPI:DATA

Command	:TRIGger:SPI:DATA <data>

	:TRIGger:SPI:DATA?
Description	Set or query the data value triggered by SPI.
Parameters	:<data> ::= 0—(2^{32}-1)
Return	The query returns an integer.
Example	:TRIGger:SPI:DATA 20 /*Set the data value to 20*/ :TRIGger:SPI:DATA? /*The query returns 20*/

4.18.6 TRIGger:SPI:MASK

Command	:TRIGger:SPI:MASK <mask> :TRIGger:SPI:MASK?
Description	Set or query the mask value when the SPI triggered.
Parameters	:<mask> ::= 0—(2^{32}-1)
Return	The query returns an integer.
Example	:TRIGger:SPI:MASK 20 /*Set the mask value to 20*/ :TRIGger:SPI:MASK? /*The query returns 20*/

4.18.7 TRIGger:SPI:ALEVel

Command	:TRIGger:SPI:ALEVel <level> :TRIGger:SPI:ALEVel?
Description	Set or query the trigger level of the clock channel when SPI triggers.
Parameters	:<level> ::= trigger level value (in V)
Return	The query returns the trigger level in scientific notation.
Example	:TRIGger:SPI:ALEVel 0.16 /*Set the trigger level to 160mV*/ :TRIGger:SPI:ALEVel? /*The query returns 1.600000e-01*/

4.18.8 TRIGger:SPI:BLEVel

Command	:TRIGger:SPI:BLEVel <level> :TRIGger:SPI:BLEVel?
Description	Set or query the trigger level of the data channel during SPI trigger.
Parameters	:<level> ::= trigger level value (in V)
Return	The query returns the trigger level in scientific notation.
Example	:TRIGger:SPI:BLEVel 0.16 /*Set the trigger level to 160mV*/ :TRIGger:SPI:BLEVel? /*The query returns 1.600000e-01*/

4.19 TRIGger:PATTern

Command List

- TRIGger:PATTern: PATTern
- TRIGger:PATTern:LEVel

4.19.1 TRIGger:PATTern:PATTern

Command	:TRIGger:PATTern:PATTern <pa_ch1>[,<pa_ch2>[,<pa_ch3> [,<pa_ch4>[,<pa_d0>…[,<pa_d15>]]]]] :TRIGger:PATTern:PATTern?			
Description	Set or query the pattern of each channel when the pattern is triggered.			
Parameters	Name	Type	Range	Default

	:<pa_ch1>	Discrete	{H|L|X|}	X
	:<pa_ch2>	Discrete	{H|L|X|}	X
	:<pa_ch3>	Discrete	{H|L|X|}	X
	:<pa_ch4>	Discrete	{H|L|X|}	X
	:<pa_D10>	Discrete	{H|L|X|}	C
	...	...	...	...
	:<pa_D43>	Discrete	{H|L|X|}	C
State	Parameters <pa_ch1> to <pa_ch4> set the pattern of the analog channels CH1 to CH4, and parameters <pa_D10> to <pa_D43> set the pattern of the digital channel. The user can send 20 parameters to set the pattern of all channels, or omit some parameters to set the pattern of some channels (the channel is omitted, the pattern status remains the same), but at least one parameter needs to be sent (the parameter sets CH1 Pattern). When less than 20 parameters are sent, the instrument defaults to set CH1 to CH4 and D10 to D43 in turn. In the parameter value range, H means high level (above the threshold level of the channel), L means low level (below the threshold level of the channel), X means ignore this channel (this channel is not part of the pattern, when all channels are set to X, the oscilloscope will not trigger).			
Return	The query returns the pattern currently set for 4 analog channels or all channels. Multiple channels are separated by commas.			
Example	:TRIGger:PATTern:PATTern H,R,L,X /*Set the code pattern of CH1 to CH4 to H, R, L, X, the pattern of other channels will not change*/ :TRIGger:PATTern:PATTern? /*The query returns H, R, L, X, X, X, X, X, X, X, X, X, X, X, X, X, X, X, X, X, X*/			

4.19.2 :TRIGger:PATTern:LEVel

Command	:TRIGger:PATTern:LEVel <chan>,<level> :TRIGger:PATTern:LEVel? <chan>			
Description	Set or query the trigger level of the specified channel when the pattern is triggered.			
Parameters	Name	Type	Range	Default
	:<level>	Integer	(-5 × VerticalScale - OFFSet) to (5 × VerticalScale - OFFSet)	0
State	This setting command is valid only when the selected source is an analog channel.			
Example	:TRIGger:PATTern:LEVel CHANnel2,0.16 /*et the trigger level of CH2 to 160mV*/ :TRIGger:PATTern:LEVel? CHANnel2 /*The query returns 1.600000e-01*/			

5. CALibrate Command Subsystem

Command List

- CALibrate:STARt
- CALibrate:STATus?
- CALibrate:QUIT

5.1 CALibrate:STARt

Command	:CALibrate:STARt

Description	The oscilloscope starts performing a self-calibration operation.
State	▷ Self-calibration operation can quickly make the oscilloscope reach the best working condition to obtain the most accurate measurement value. ▷ Before performing a self-calibration, make sure that no signals are connected to all channels until the self-calibration operation ends. ▷ During the self-calibration process, most of the key functions have been disabled.

5.2 CALibrate:STATus?

Command	:CALibrate:STATus?
Description	Returns the status of the current calibration.

5.3 CALibrate:QUIT

Command	:CALibrate:QUIT
Description	Exit the self-calibration operation at any time.

6. MATH Command Subsystem

Command List

- MATH:DISPlay
- MATH:OPERator
- MATH:SOURce1
- MATH:SOURce2
- MATH:SCALe
- MATH:OFFSet
- MATH:FFT:SOURce
- MATH:FFT:WINDow
- MATH:FFT:UNIT
- MATH:FFT:HSCale
- MATH:FFT:HCENter

6.1 MATH:DISPlay

Command	:MATH:DISPlay <bool> :MATH:DISPlay?
Description	Turn the math operation function on or off or query the status of the math operation function
Parameters	:<bool> ::= {{1 | ON} | {0 | OFF}

Return	The query returns ON,OFF.
Example	:MATH:DISPlay ON /*Turn on the mathematical operation function*/ :MATH:DISPlay? /*Query returns ON*/

6.2 MATH:OPERator

Command	:MATH:OPERator <type> :MATH:OPERator?
Description	Sets or queries the operator of a mathematical operation.
Parameters	:<type> ::= < ADD| SUBTract| MULTiply| DIVision| FFT>
Return	The query returns ADD,SUBTract,MULTiply,DIVision,FFT
Example	:MATH:OPERator ADD /*Set the mathematical operator to add operation*/ :MATH:OPERator? /*Query returns ADD*/

6.3 MATH:SOURce1

Command	:MATH:SOURce1 <source> :MATH:SOURce1?
Description	Set or query the source or source A of algebraic operations.
Parameters	:<source> ::= <CHANnel1| CHANnel2| CHANnel3| CHANnel4>
Return	The query returns CHANnel1| CHANnel2| CHANnel3| CHANnel4
Example	:MATH:SOURce1 CHANnel1 /*Set the source A of algebraic operation to CHANnel1*/ :MATH:SOURce1? /*The query returns CHANnel1*/

6.4 MATH:SOURce2

Command	:MATH:SOURce2 <source> :MATH:SOURce2?
Description	Set or query the source B of the algebraic operation / combination operation.
Parameters	:<source> ::= <CHANnel1| CHANnel2| CHANnel3| CHANnel4>
Return	The query returns CHANnel1| CHANnel2| CHANnel3| CHANnel4
Example	:MATH:SOURce2 CHANnel1 /*Set the source A of algebraic operation to CHANnel1*/ :MATH:SOURce2? /*The query returns CHANnel1*/

6.5 MATH:SCALe

Command	:MATH:SCALe <value> :MATH:SCALe?
Description	Set or query the vertical scale of the operation result. The unit is related to the currently selected operator and the unit selected by the source.
Parameters	:<value> ::= vertical scale (in 1-2-5 sequence, unit is V)
Return	The query returns the vertical scale of the current operation result in scientific notation.
Example	:MATH:SCALe 2 /*Set the vertical scale to 2V*/ :MATH:SCALe? /*The query returns 2.000000e + 00*/

6.6 MATH:OFFSet

Command	:MATH:OFFSet <value> :MATH:OFFSet?
Description	Sets or queries the vertical offset of the operation result.
Parameters	:<value> ::= offset value (in V)
Return	The query returns the vertical offset of the current operation result in scientific notation.
Example	:MATH:OFFSet 2 /*Set vertical offset to 2V*/ :MATH:OFFSet? /*The query returns 2.000000e + 00*/

6.7 MATH:FFT:SOURce

Command	:MATH:FFT:SOURce <source> :MATH:FFT:SOURce?
Description	Set or query the source of FFT operation / filter.
Parameters	:<source> ::= < CHANnel1|CHANnel2|CHANnel3|CHANnel4>
Return	The query returns CHANnel1, CHANnel2, CHANnel3, or CHANnel4.
Example	:MATH:FFT:SOURce CHANnel1 /*Set the source of FFT operation to CH1*/ :MATH:FFT:SOURce? /*The query returns CHAnel1*/

6.8 MATH:FFT:WINDow

Command	:MATH:FFT:WINDow <window> :MATH:FFT:WINDow?
Description	Sets or queries the window function of the FFT operation.
Parameters	:<window> ::= < RECTangle| HANNing| HAMMing| BLACkman| TRIangle| FLATtop>
State	▷Using the window function can effectively reduce the effect of spectral leakage. ▷Each window function is suitable for measuring different waveforms. You need to choose according to the measured waveform and its characteristics.
Return	The query returns RECTangle,HANNing,HAMMing,BLACkman,TRIangle,FLATtop
Example	:MATH:FFT:WINDow RECTangle /*Set the window function for FFT operation to RECTangle*/ :MATH:FFT:WINDow? /*The query returns RECTangle*/

6.9 MATH:FFT:UNIT

Command	:MATH:FFT:UNIT <unit> :MATH:FFT:UNIT?
Description	Sets or queries the vertical unit of the FFT operation result.
Parameters	:<unit> ::= < VRMS| DB>
Return	Query returns VRMS, DB
Example	:MATH:FFT:UNIT DB /*Set the vertical unit of the FFT operation result to DB*/ :MATH:FFT:UNIT? /*The query returns DB*/

6.10 MATH:FFT:HSCale

Command	:MATH:FFT:HSCale <hscale>

	:MATH:FFT:HSCale?
Description	Set or query the horizontal scale of the FFT operation result. The default unit is Hz.
Parameters	:<hscale> ::= <125000|250000|625000|1250000>
State	You can observe the detailed information of the spectrum by reducing the horizontal scale.
Return	The query returns the current horizontal scale in scientific notation.
Example	:MATH:FFT:HSCale 125000 /*Set the horizontal scale of the FFT calculation result to 125kHz*/ :MATH:FFT:HSCale? /*The query returns 1.25000e + 05*/

6.11 MATH:FFT:HCENter

Command	:MATH:FFT:HCENter <center> :MATH:FFT:HCENter?
Description	Set or query the center frequency of the FFT operation result, that is, the frequency corresponding to the horizontal center of the screen.
Parameters	:<center> ::= center frequency (in Hz)
Return	The query returns the current center frequency in scientific notation.
Example	:MATH:FFT:HCENter 10000000 /*Set the center frequency of the FFT operation result to 10MHz*/ :MATH:FFT:HCENter? /*The query returns 1.000000e + 07*/

7. WAVeform Command Subsystem

Command List

- WAVeform:DATA:ALL?

7.1 WAVeform:DATA:ALL?

Command	:WAVeform:DATA:ALL?
Description	Read waveform data.
Return	Return the waveform data packet containing the data header in the form of a string.
Remark	The first time this command is issued, the analysis of data[x] is as follows: data[0]-data[1] (2 digits): #9 data[2]-data[10](9 digits):The byte length of the current packet. data[11]-data[19](9 digits): The total length of bytes representing the amount of data. data[20]-data[28](9 digits): The byte length of the uploaded data. data[29](1 digit): The current running status. data[30](1 digit) : Trigger status. data[31-34] (4 digits) : Offset of channel 1 data[35-38] (4 digits): Offset of channel 2 data[39-42] (4 digits): Offset of channel 3 data[43-46] (4 digits): Offset of channel 4 data[47]-data[53](7 digit): Voltage of channel 1 data[54]-data[60](7 digits): Voltage of channel 2 data[61]-data[67](7 digits): Voltage of channel 3 data[68]-data[74](7 digits): Voltage of channel 4 data[75]-data[78] (4 digit): Channel enable of channel (1-4) data[79]-data[87] (9 digits): Sampling rate. data[88]-data[93] (6digits): Sampling multiple.

	data[94]-data[102] (9 digits): Display trigger time of current frame. data[103]-data[111] (9 digits): The current frame displays the start point of the data acquisition start time point. data[112]-data[127] (16digits): Reserved bit. This command is issued again before this data is read to analyze the waveform data data[x] as follows: data[0]-data[1] (2 digits): #9 data[2]-data[10] (9 digits): Indicates the byte length of the current data packet. data[11]-data[19] (9 digits): the total length of bytes representing the amount of data, data[20]-data[28] (9 digits): Indicates the byte length of the uploaded data. data[29]-data[x]: indicates the waveform data corresponding to the current data header.

8. DISPlay Command Subsystem

Command List

- DISPlay:TYPE
- DISPlay:WBRightness
- DISPlay:GRID
- DISPlay:GBRightness

8.1 DISPlay:TYPE

Command	:DISPlay:TYPE <type> :DISPlay:TYPE?
Description	Sets or queries how waveforms are displayed on the screen.
Parameters	:<type> ::= < VECTors| DOTS>
State	▷ VECTors: Displayed by connecting lines between sampling points. This mode provides the most realistic waveform in most cases. Easily view the steep edges of a waveform, such as a square wave. ▷ DOTS: Display sampling points directly. You can see each sampling point visually and use the cursor to measure the X and Y values of that point.
Return	The query returns VECT or DOTS.
Example	:DISPlay:TYPE DOTS /*Selection point display method*/ :DISPlay:TYPE? /*The query returns DOTS*/

8.2 DISPlay:WBRightness

Command	:DISPlay:WBRightness <value> :DISPlay:WBRightness?
Description	Set or query the brightness of the waveform display on the screen.
Parameters	:<value> ::= one number of 0 to 100
Return	The query returns an integer
Example	:DISPlay:WBRightness 50 /*Set the waveform brightness to 50%*/ :DISPlay:WBRightness? /*The query returns 50*/

8.3 DISPlay:GRID

Command	:DISPlay:GRID <type> :DISPlay:GRID?
Description	Sets or queries the type of grid displayed on the screen.
Parameters	:<type> ::= < DOTTed| REAL>
State	DOTTed: dot grid REAL: line grid
Return	The query returns DOTTed, REAL
Example	:DISPlay:GRID DOTTed /*set the background grid to DOTTed */ :DISPlay:GRID? /*The query returns DOTTed*/

8.4 DISPlay:GBRightness

Command	:DISPlay:GBRightness <value> :DISPlay:GBRightness?
Description	Sets or queries the brightness of the screen grid.
Parameters	:<value> ::= one number of 0 to 100
Return	The query returns an integer
Example	:DISPlay:WBRightness 50 /*Set the screen grid brightness to 50%*/ :DISPlay:WBRightness? /*The query returns 50*/

9. CURSor Command Subsystem

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

Command List

- CURSor:MODE
- CURSor:MANual:TYPE
- CURSor:MANual:SOURce
- CURSor:MANual:AX
- CURSor:MANual:AXValue?
- CURSor:MANual:AY
- CURSor:MANual:AYValue?
- CURSor:MANual:BX
- CURSor:MANual:BXValue?
- CURSor:MANual:BY
- CURSor:MANual:BYValue?

- CURSor:TRACk:SOURcea
- CURSor:TRACk:SOURceb
- CURSor:TRACk:AX
- CURSor:TRACk:AXValue?
- CURSor:TRACk:AY?
- CURSor:TRACk:AYValue?
- CURSor:TRACk:BX
- CURSor:TRACk:BXValue?
- CURSor:TRACk:BY?
- CURSor:TRACk:BYValue?

9.1 CURSor:MODE

Command	:CURSor:MODE <type> :CURSor:MODE?
Description	Set or query the cursor measurement mode.
Parameters	:<type> ::= < OFF| MANual| TRACk >
State	MANual: Turn on the manual cursor measurement mode. TRACk: Turn on the cursor tracking measurement mode.
Return	The query returns OFF, MANual, TRACK
Example	:CURSor:MODE TRACK /*Select TRACK cursor measurement mode*/ :CURSor:MODE? /*The query returns TRACK*/

9.2 CURSor:MANual:TYPE

Command	:CURSor:MANual:TYPE <type> :CURSor:MANual:TYPE?
Description	Set or query the cursor type for manual cursor measurement.
Parameters	:<type> ::= <X | Y | XY>
State	▷ X: Select X type cursor. The X-shaped cursor is a vertical solid line (Cursor A) and a vertical dotted line (Cursor B). It is usually used to measure time parameters. ▷ Y: Select the Y cursor. The Y cursor is a horizontal solid line (Cursor A) and a horizontal dotted line (Cursor B). It is usually used to measure voltage parameters. ▷ XY: Select the XY cursor.
Return	The query returns X, Y, XY.
Example	:CURSor:MANual:TYPE X /*Select X type cursor*/ :CURSor:MANual:TYPE? /*The query returns X*/

9.3 CURSor:MANual:SOURce

Command	:CURSor:MANual:SOURce <source>

	:CURSor:MANual:SOURce?
Description	Set or query the channel source for manual cursor measurement.
Parameters	:<source> ::= < CHANnel1|CHANnel2| MATH>
Return	The query returns CHANnel1, CHANnel2, MATH
Example	:CURSor:MANual:SOURce CHANnel1 /*Set the channel source to CH1*/ :CURSor:MANual:SOURce? /*The query returns CHANnel1*/

9.4 CURSor:MANual:AX

Command	:CURSor:MANual:AX <value> :CURSor:MANual:AX?
Description	Set or query the horizontal position of cursor A when manual cursor measurement is performed.
Parameters	:<type> ::= one number of 0 to 770
State	The horizontal and vertical positions of the cursor are defined by the pixel coordinates of the screen.
Return	:CURSor:MANual:AX 200 /*Set the horizontal position of cursor A to 200*/ :CURSor:MANual:AX? /*The query returns 200*/

9.5 CURSor:MANual:AXValue?

Command	:CURSor:MANual:AXValue?
Description	Query the X value at cursor A during manual cursor measurement. The unit is determined by the currently selected horizontal unit.
Return	The query returns the X value at the current cursor A in scientific notation.
Example	:CURSor:MANual:AXValue? /*The query returns -4.000000e-06*/

9.6 CURSor:MANual:AY

Command	:CURSor:MANual:AY <value> :CURSor:MANual:AY?
Description	Set or query the vertical position of cursor A when manual cursor measurement.
Parameters	:<type> ::= 0--400
State	The horizontal and vertical positions of the cursor are defined by the pixel coordinates of the screen.
Return	The query returns an integer among 0 and 400.
Example	:CURSor:MANual:AY 200 /*Set the vertical position of cursor A to 200*/ :CURSor:MANual:AY? /*The query returns 200*/

9.7 CURSor:MANual:AYValue?

Command	:CURSor:MANual:AYValue?
Description	Query the Y value at cursor A during manual cursor measurement. The unit is determined by the currently selected vertical unit.
Return	The query returns the Y value at the current cursor A in scientific notation.
Example	:CURSor:MANual:AYValue? /*The query returns 2.000000e + 00*/

9.8 CURSor:MANual:BX

Command	:CURSor:MANual:BX <value> :CURSor:MANual:BX?
Description	Set or query the horizontal position of cursor B when manual cursor measurement is performed.
Parameters	:<type> ::= 0-770
State	The horizontal and vertical positions of the cursor are defined by the pixel coordinates of the screen.
Return	The query returns an integer among 0 and 770.
Example	:CURSor:MANual:BX 200 /*Set the horizontal position of cursor B to 200*/ :CURSor:MANual:BX? /*The query returns 200*/

9.9 CURSor:MANual:BXValue?

Command	:CURSor:MANual:BXValue?
Description	When querying the manual cursor measurement, the X value at cursor B. The unit is determined by the currently selected horizontal unit.
Return	The query returns the X value at the current cursor B in scientific notation.
Example	:CURSor:MANual:BXValue? /*The query returns 4.000000e-06*/

9.10 CURSor:MANual:BY

Command	:CURSor:MANual:BY <value> :CURSor:MANual:BY?
Description	Set or query the vertical position of cursor B when manual cursor measurement is performed.
Parameters	:<type> ::= 0-400
State	The horizontal and vertical positions of the cursor are defined by the pixel coordinates of the screen.
Return	The query returns an integer among 0 and 400.
Example	:CURSor:MANual:BY 200 /*Set the vertical position of cursor B to 200*/ :CURSor:MANual:BY? /*The query returns 200*/

9.11 CURSor:MANual:BYValue?

Command	:CURSor:MANual:BYValue?
Description	Query the Y value at cursor B during manual cursor measurement. The unit is determined by the currently selected vertical unit.
Parameters	The query returns the Y value at the current cursor B in scientific notation.
Example	:CURSor:MANual:BYValue? /*The query returns 2.000000e + 00*/

9.12 CURSor:TRACk:SOURcea

Command	:CURSor:TRACk:SOURcea <source> :CURSor:TRACk:SOURcea?
Description	Set or query the channel source of cursor A measurement during cursor tracking measurement.
Parameters	:<source> ::= < CHANnel1|CHANnel2|CHANnel3|CHANnel4| MATH >

Return	The query returns CHANnel1, CHANnel2, CHANnel3, CHANnel4 or MATH.
Example	:CURSor:TRACk:SOURcea CHANnel1 /*Set the channel source to CH1*/ :CURSor:TRACk:SOURcea? /*The query returns CHANnel1*/

9.13 CURSor:TRACk:SOURceb

Command	:CURSor:TRACk:SOURceb <source> :CURSor:TRACk:SOURceb?
Description	Set or query the channel source of cursor B measurement during cursor tracking measurement.
Parameters	:<source> ::= < CHANnel1|CHANnel2|CHANnel3|CHANnel4| MATH >
Return	The query returns CHANnel1, CHANnel2, CHANnel3, CHANnel4 or MATH.
Example	:CURSor:TRACk:SOURceb CHANnel1 /*Set the channel source to CH1*/ :CURSor:TRACk:SOURceb? /*The query returns CHANnel1*/

9.14 CURSor:TRACk:AX

Command	:CURSor:TRACk:AX <value> :CURSor:TRACk:AX?
Description	Set or query the horizontal position of cursor A when cursor tracking measurement.
Parameters	:<type> ::= 0-770
State	The horizontal and vertical positions of the cursor are defined by the pixel coordinates of the screen.
Return	The query returns an integer among 0 and 770.
Example	:CURSor:TRACk:AX 200 /*Set the horizontal position of cursor A to 200*/ :CURSor:TRACk:AX? /*The query returns 200*/

9.15 CURSor:TRACk:AXValue?

Command	:CURSor:TRACk:AXValue?
Description	Query the X value at cursor A during track cursor measurement. The default unit is second.
Return	The query returns the X value at the current cursor A in scientific notation.
Example	:CURSor:TRACk:AXValue? /*The query returns -4.000000e-06*/

9.16 CURSor:TRACk:AY?

Command	:CURSor:TRACk:AY?
Description	Query the vertical position of cursor A during cursor tracking measurement.
Return	The query returns an integer.
Example	:CURSor:TRACk:AY? /*The query returns 288*/

9.17 CURSor:TRACk:AYValue?

Command	:CURSor:TRACk:AYValue?
Description	Query the Y value at cursor A during track cursor measurement. The unit is determined by the currently selected channel’s unit.
Return	The query returns the Y value at the current cursor A in scientific notation.
Example	:CURSor:TRACk:AYValue? /*The query returns -4.000000e-01*/

9.18 CURSor:TRACk:BX

Command	:CURSor:TRACk:BX <value> :CURSor:TRACk:BX?
Description	Set or query the horizontal position of cursor B when cursor tracking measurement.
Parameters	:<type> ::= 0-770
State	The horizontal and vertical positions of the cursor are defined by the pixel coordinates of the screen.
Return	The query returns an integer among 0 and 770.
Example	:CURSor:TRACk:BX 200 /*Set the horizontal position of cursor B to 200*/ :CURSor:TRACk:BX? /*The query returns 200*/

9.19 CURSor:TRACk:BXValue?

Command	:CURSor:TRACk:BXValue?
Description	Query the X value at cursor B during track cursor measurement. The default unit is second.
Return	The query returns the X value at the current cursor B in scientific notation.
Example	:CURSor:TRACk:BXValue? /*The query returns -4.000000e-06*/

9.20 CURSor:TRACk:BY?

Command	:CURSor:TRACk:BY?
Description	Query the vertical position of cursor B during cursor tracking measurement
Return	The query returns an integer.
Example	:CURSor:TRACk:BY? /*The query returns 288*/

9.21 CURSor:TRACk:BYValue?

Command	:CURSor:TRACk:BYValue?
Description	Query the Y value at cursor B during track cursor measurement. The unit is determined by the currently selected channel’s unit.
Return	The query returns the Y value at the current cursor B in scientific notation.
Example	:CURSor:TRACk:BYValue? /*The query returns -4.000000e-01*/

10. MEASure Command Subsystem

Command List

- MEASure:ENABle
- MEASure:SOURce
- MEASure:ADISplay
- MEASure: CHANnel<n>:ITEM
- MEASure:GATE:ENABle

- MEASure:GATE:AY
- MEASUre:GATE:BY

10.1 MEASure:ENABle

Command	:MEASure:ENABle <bool> :MEASure:ENABle?
Description	Set or query the measurement function status
Parameters	:<bool> ::= {{1 | ON} | {0 | OFF}
Return	The query returns ON,OFF
Example	:MEASure:ENABle ON /*turn on measurement function*/ :MEASure:ENABle? /*Query returns ON*/

10.2 MEASure:SOURce

Command	:MEASure:SOURce <source> :MEASure:SOURce?
Description	Set or query the source of the current measurement parameters.
Parameters	:<source> ::= < CHANnel1|CHANnel2|CHANnel3|CHANnel4| MATH>
Return	The query returns CHANnel1, CHANnel2, CHANnel3, CHANnel4,MATH
Example	:MEASure:SOURce CHANnel1 /*Set parameter measurement source to CH1*/ :MEASure:SOURce? /*Query returns CHANnel1*/

10.3 MEASure:ADISplay

Command	:MEASure:ADISplay <bool> :MEASure:ADISplay?
Description	Turn all measurements on or off, or query the current status of all measurements.
Parameters	:<bool> ::= {{1 | ON} | {0 | OFF}
Return	The query returns ON,OFF.
Example	:MEASure:ADISplay ON /*Turn on all measurements*/ :MEASure:ADISplay? /*Query returns ON*/

10.4 MEASure: CHANnel<n>:ITEM

Command	:MEASure: CHANnel<n>:ITEM <type> :MEASure: CHANnel<n>:ITEM?
Description	Query the measurement result of the specified parameter
Parameters	:<n>::= {1 | 2 | 3 | 4} :<type>::=< MAX,VMIN,VPP,VTOP,VBASe,VAMP,VAVG,VRMS,OVERshoot,PREShoot,MARea,MPARea,PERiod,FREQuency,RTIMe,FTIMe,PWIDth,NWIDth,PDUTy,NDUTy,RDELay,FDELay,RPHase,FPHase,TVMAX,TVMIN,PSLEWrate,NSLEWrate,VUPper, VMID,VLOWer,VARIance,PVRMS,PPULses,NPULses,PEDGes,NEDGes>
Return	The query returns
Example	:MEASure:CHANnel1:ITEM? VPP /*The query returns VPP 3.600e-01 */

10.5 MEASure:GATE:ENABle

Command	:MEASure:GATE:ENABle <bool> :MEASure:GATE:ENABle?
Description	Set or query the current gate control status
Parameters	:<bool> ::= {{1 | ON} | {0 | OFF}
Return	The query returns ON,OFF
Example	:MEASure:GATE:ENABle ON /*Turn on the gate function*/ :MEASure:GATE:ENABle? /*Query returns ON*/

10.6 MEASure:GATE:AY

Command	:MEASure:GATE:AY <value> :MEASure:GATE:AY?
Description	Set or query the value of cursor A.
Parameters	:<value> ::= 0--400
Return	The query returns an integer.
Example	:MEASure:GATE:AY 100 /*Set the value of cursor A*/ :MEASure:GATE:AY? /*Query returns 100*/

10.7 MEASUre:GATE:BY

Command	:MEASure:GATE:BY <value> :MEASure:GATE:BY?
Description	Set or query the value of cursor B.
Parameters	:<value> ::= 0--400
Return	The query returns an integer.
Example	:MEASure:GATE:BY 100 /*Set the value of cursor B*/ :MEASure:GATE:BY? /*Query returns 100*/

11. MASK Command Subsystem

Command List

- MASK:EANBle
- MASK:SOURce
- MASK:MDISplay
- MASK:SOOutput
- MASK:OUTPut
- MASK:X
- MASK:Y
- MASK:CREate

11.1 MASK:EANBle

Command	:MASK:EANBle <bool> :MASK:EANBle?
Description	Turn the pass / fail test function on or off, or query the status of the pass / fail test function.
Parameters	:<bool> ::= {{1 | ON} | {0 | OFF}
Return	The query returns ON,OFF
Example	:MASK:EANBle ON /*Turn on the pass / fail test function*/ :MASK:EANBle? /*The query returns ON*/

11.2 MASK:SOURce

Command	:MASK:SOURce <source> :MASK:SOURce?
Description	Set or query the measurement source of the pass / fail test.
Parameters	:<source> ::= < CHANnel1|CHANnel2|MATH>
Return	The query returns CHANnel1, CHANnel2, MATH
Example	:MASK:SOURce CHANnel1 /*Set the measurement source of the pass / fail test to CH1*/ :MASK:SOURce? /*The query returns CHANnel1*/

11.3 MASK:MDISplay

Command	:MASK:MDISplay <bool> :MASK:MDISplay?
Description	When the pass / fail test is turned on, turn statistics on or off, or query the status of the statistics.
Parameters	:<bool> ::= {{1 | ON} | {0 | OFF}
Return	The query returns ON,OFF.
Example	:MASK:MDISplay ON /*Open statistics*/ :MASK:MDISplay?/*The query returns ON*/

11.4 MASK:Output

Command	:MASK:OUTPut <bool> :MASK:OUTPut?
Description	Turn output on or off, or query output stop status.
Parameters	:<bool> ::= {{1 | ON} | {0 | OFF}
State	▷On: When a failed waveform is detected, the oscilloscope will stop the test and enter the "STOP" state. At this time, the screen keeps displaying the measurement result (if the display is turned on), and the rear panel [Trigger Out] (if enabled) outputs only one pulse. ▷Off: Even if a failed waveform is detected, the oscilloscope will continue the test, and the test results on the screen are continuously updated. The [Trigger Out] on the rear panel will output a pulse each time a failed waveform is detected.
Return	The query returns ON,OFF
Example	:MASK:OUTPut ON /*Stop the output*/ :MASK:OUTPut? /*The query returns ON*/

11.5 MASK:SOOutput

Command	:MASK:SOOutput <bool> :MASK:SOOutput?
Description	Turn on or off the sound prompt when the test fails output, or query the status of the sound prompt.
Parameters	:<bool> ::= {{1 | ON} | {0 | OFF}
State	▷Off: When a failed waveform is detected, it will be displayed and output, but the buzzer will not alarm. ▷On: When a failed waveform is detected, it will be displayed and output, and the buzzer will give an audible alarm (not related to the sound switch state)
Return	The query returns ON,OFF
Example	:MASK:SOOutput ON /*Turn on the sound prompt*/ :MASK:SOOutput? /*The query returns ON*/

11.6 MASK:X

Command	:MASK:X <value> :MASK:X?
Description	Set or query the level adjustment parameter in the pass / fail test rule. The default unit is div.
Parameters	:<value> ::= 0.02 - 4
Return	The query returns the current level adjustment parameter in scientific notation.
Example	:MASK:X 0.28 /*Set the level adjustment parameter to 0.28div*/ :MASK:X? /*The query returns 2.800000e-01*/

11.7 MASK:Y

Command	:MASK:Y <value> :MASK:Y?
Description	Sets or queries the vertical adjustment parameter in the pass / fail test rule. The default unit is div.
Parameters	:<value> ::= 0.04 -- 5.12
Return	The query returns the current vertical adjustment parameter in scientific notation.
Example	:MASK:Y 0.36 /*Set the vertical adjustment parameter to 0.36div*/ :MASK:Y? /*The query returns 3.600000e-01*/

11.8 MASK:CREate

Command	:MASK:CREate
Description	Create a pass / fail rule with the currently set horizontal and vertical adjustment parameters.
State	This command is valid only when the pass / fail test function is turned on (: MASK: ENABle) and is not running (: MASK: OPERate)

12. SYSTem Command Subsystem

Command List

- SYSTem:GAM?
- SYSTem:RAM?
- SYSTem:PON
- SYSTem:LANGuage
- SYSTem:LOCKed
- SETUp:ALL?

12.1 SYSTem:GAM?

Command	:SYSTem:GAM?
Description	Query the number of grids in the horizontal direction of the instrument screen.
Return	The query returns 12.

12.2 SYSTem:RAM?

Command	:SYSTem:RAM?
Description	Query the number of analog channels of the instrument.
Return	The query returns 4.

12.3 SYSTem:PON

Command	:SYSTem:PON <value> :SYSTem:PON?
Description	Set or query the type of configuration called when the oscilloscope is powered on again.
Parameters	:<value> ::= < LATest| DEFault>
Return	The query returns LATest or DEFault.
Example	:SYSTem:PON LATest /*Set the last value to be called when the oscilloscope is powered on again*/ :SYSTem:PON? /*Query returns LATest*/

12.4 SYSTem:LANGuage

Command	:SYSTem:LANGuage <value> :SYSTem:LANGuage?
Description	Set or query the language displayed by the system.
Parameters	:<value> ::= < ENGLish| SCHinese>
Return	The query returns ENGLish, SCHinese
Example	:SYSTem:LANGuage SCHinese /*Set the system language to Simplified Chinese*/ :SYSTem:LANGuage? /*Query returns SCHinese*/

12.5 SYSTem:LOCKed

Command	:SYSTem:LOCKed <bool>

	:SYSTem:LOCKed?
Description	Turn the keyboard lock function on or off, or query the status of the keyboard lock function.
Parameters	:<bool> ::= {{1 | ON} | {0 | OFF}
Return	The query returns ON,OFF.

12.6 SETUp:ALL?

Command	:SETUp:ALL?
Description	Get all the states needed to boot up at once.
Return	A string containing each setting, and each status is separated by ";".
Remark	Channel enable: <Detailed view CHANnel<n>:DISPlay> Channel coupling: <Detailed view CHANnel<n>:COUPling> Channel bandwidth limit: <Detailed view CHANnel<n>:BWLimit> Probe ratio: <Detailed view CHANnel<n>:PROBe> Voltage gear: <> Channel offset: The offset (one value) of the waveform relative to the center line (the value is zero, the upward is positive, and the downward is negative). Large divisions represent 25 values). For example, the offset value of channel 1 is 75, which means that the center line is offset by three large divisions. Channel inversion: <Detailed view CHANnel<n>:INVert> Running status: <Detailed view RUNning> Acquisition mode: <Detailed view ACQuire:MODe> Collection type: <Detailed view ACQuire:TYPE> Trigger method: <Detailed view TRIGger:SWEep> Time base value: <Detailed view TIMebase:SCALe> (Placeholder): Sampling rate: The returned value represents the current sampling rate. Storage depth: <For details, see ACQuire:POINts> Trigger type: <Detailed view TRIGger:MODE> (Placeholder): Edge trigger source: <Detailed view TRIGger:EDGe:SOURce> Edge trigger level: <See TRIGger:EDGe:LEVel for details> Edge trigger polarity: <See TRIGger:EDGe:SLOPe for details> Pulse width trigger source: <See TRIGger:PULSe:SOURce for details> Pulse width trigger level: <See TRIGger:PULSe:LEVel for details> Pulse width trigger polarity: <See TRIGger:PULSe:POLarity for details> Pulse width trigger condition: <See TRIGger:PULSe:WHEN for details> Pulse width trigger width: <For details, see TRIGger:PULSe:WIDth> Timeout trigger source: <For details, see TRIGger:TIMeout:SOURce> Timeout trigger level: <For details, see TRIGger:TIMeout:LEVel> Timeout trigger polarity: <For details, see TRIGger:TIMeout:POLarity> Timeout trigger width: <For details, see TRIGger:TIMeout:WIDth> Slope trigger source: <For details, see TRIGger:SLOPe:SOURce> Slope trigger level a: <See TRIGger:SLOPe:ALEVel for details> Slope trigger level b: <For details, see TRIGger:SLOPe:BLEVel> Slope trigger polarity: <See TRIGger:SLOPe:POLarity for details> Conditions for slope triggering: <For details, see TRIGger:SLOPe:WHEN> Slope trigger width: <For details, see TRIGger:SLOPe:WIDth> Video trigger source: <For details, see TRIGger:TV:SOURce> Video trigger level: <See TRIGger:VIDeo:LEVel for details> Video trigger polarity: <See TRIGger:TV:POLarity for details> Video trigger standard: <Detailed view TRIGger:TV:STANdard> Video trigger synchronization: <For details, see TRIGger:TV:MODE> The number of lines triggered by the video: <See TRIGger:TV:LINE for details>

	Window trigger source: <Detailed view TRIGger:WINDOw:SOURce> Window trigger level a: <See TRIGger:WINDOw:ALEVel for details> Window trigger level b: <See TRIGger:WINDOw:BLEVel for details> Interval trigger source: <For details, see TRIGger:INTERVAl:SOURce> Interval trigger edge: <See TRIGger:INTERVAl:SLOp for details> Interval trigger condition: <See TRIGger:INTERVAl:WHEN for details> Interval trigger time: <See TRIGger:INTERVAl:TIME for details> Interval trigger level: <See TRIGger:INTERVAl:ALEVel for details> Runt trigger source: <For details, see TRIGger:UNDER_Am:SOURce> Runt trigger polarity: <See TRIGger:UNDER_Am:POLarity for details> Runt trigger condition: <See TRIGger:UNDER_Am:WHEN for details> Runt trigger time: <Detailed view TRIGger:UNDER_Am:TIME> Runt trigger level a: <See TRIGger:UNDER_Am:ALEVel for details> Runt trigger level b: <See TRIGger:UNDER_Am:BLEVel for details> Serial port trigger source: <Detailed view TRIGger:UART:SOURce> Serial port trigger conditions: <For details, see TRIGger:UART:CONdition> Serial port trigger baud rate: <For details, see TRIGger:UART:BAUd> Serial port trigger level: <For details, see TRIGger:UART:ALEVel> Serial port trigger data: <For details, see TRIGger:UART:DATA> Serial port trigger data bit width: <For details, see TRIGger:UART:WIDTh> Serial port trigger stop bit: <For details, see TRIGger:UART:STOP> Serial port trigger polarity: <For details, see TRIGger:UART:PARIty> CAN trigger source: <For details, see TRIGger:CAN:SOURce> CAN trigger idle level: <Detailed view TRIGger:CAN:IDLe> CAN trigger baud rate: <Detailed view TRIGger:CAN:BAUd> CAN trigger condition: <Detailed view TRIGger:CAN:CONdition> CAN trigger data: <Detailed view TRIGger:CAN:DATA> CAN trigger ID: <see TRIGger:CAN:ID for details> CAN trigger dlc: <see TRIGger:CAN:DLC for details> CAN trigger level: <For details, see TRIGger:CAN:ALEVel> LIN trigger source: <For details, see TRIGger:LIN:SOURce> LIN trigger idle level: <See TRIGger:LIN:IDLe for details> LIN trigger baud rate: <For details, see TRIGger:LIN:BAUd> LIN trigger condition: <Detailed view TRIGger:LIN:CONdition> LIN trigger ID: <see TRIGger:LIN:ID for details> LIN trigger data: <For details, see TRIGger:LIN:DATA> LIN trigger level: <For details, see TRIGger:LIN:ALEVel> The channel source of the IIC trigger data line: <For details, see TRIGger:IIC:SDA:SOURce> The channel source of the IIC trigger clock line: <For details, see TRIGger:IIC:SCL:SOURce> IIC trigger condition: <For details, see TRIGger:IIC:CONdition> IIC trigger address: <For details, see TRIGger:IIC:ADDer> IIC trigger data: <For details, see TRIGger:IIC:DATA> The trigger level of the IIC trigger clock channel: <For details, see TRIGger:IIC:ALEVel> Trigger level of IIC trigger data channel: <For details, see TRIGger:IIC:BLEVel> The channel source of the SPI trigger data line: <For details, see TRIGger:SPI:SDA:SOURce> The channel source of the SPI trigger clock line: <For details, see TRIGger:SPI:SCL:SOURce> SPI trigger clock edge type: <For details, see TRIGger:SPI:SCK> SPI trigger data bit width: <For details, see TRIGger:SPI:WIDth> SPI trigger data: <See TRIGger:SPI:DATA for details> SPI trigger mask data: <For details, see TRIGger:SPI:MASK> The trigger level of the SPI trigger clock channel: <For details, see

	TRIGger:SPI:ALEVel> The trigger level of the SPI trigger data channel: <For details, see TRIGger:SPI:BLEVel> Pattern triggered by pattern: <For details, see TRIGger:LOGIc:POLarity> Conditions for pattern triggering: <For details, see TRIGger:LOGIc:WHEN> Pattern trigger width: <For details, see TRIGger:LOGIc:TIME> Pattern trigger CH1 level: <For details, see TRIGger:LOGIc:ALEVel> Pattern trigger CH2 level: <For details, see TRIGger:LOGIc:BLEVel> The level of pattern trigger CH3: <For details, see TRIGger:LOGIc:CLEVel> Pattern trigger CH4 level: <For details, see TRIGger:LOGIc:DLEVel>

13. DDS Command Subsystem

Command List

- DDS:SWITch
- DDS:TYPE
- DDS:FREQ
- DDS:AMP
- DDS:OFFSet
- DDS:DUTY
- DDS:WAVE:MODE
- DDS:MODE:TYPE
- DDS:MODE:WAVE:TYPE
- DDS:MODE:FREQ
- DDS:MODE:DEPThordeviation
- DDS:BURSt:SWITch
- DDS:BURSt:TYPE
- DDS:BURSt:CNT
- DDS:BURSt:TRIGger

13.1 DDS:SWITch

Command	:DDS:SWITch <bool> :DDS:SWITch?
Description	Set or query the signal source status
Parameters	:<bool> ::= {{1 | ON} | {0 | OFF}
Return	The query returns ON,OFF

Example	:DDS:SWITch ON /*Turn on the signal source*/ :DDS:SWITch? /*The query returns ON*/

13.2 DDS:TYPE

Command	:DDS:TYPE <type> :DDS:TYPE?
Description	Set or query the signal source output wave type
Parameters	:<type> ::= { SINE| SQUAre| RAMP| EXP| NOISe| DC| ARB1 | ARB2| ARB3| ARB4}
Return	The 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? /*The query returns SINE*/

13.3 DDS:FREQ

Command	:DDS:FREQ <freq> :DDS:FREQ?
Description	Set or query the frequency of the source signal
Parameters	:<freq> ::= unit Hz
Return	Query returns frequency in scientific notation
Example	:DDS:FREQ 1000 /*Set the signal source waveform frequency to 1khz*/ :DDS:FREQ? /*The query returns 1.00000e+03*/

13.4 DDS:AMP

Command	:DDS:AMP <amp> :DDS:AMP?
Description	Set or query the amplitude of the signal source
Parameters	:<amp> ::= unit V
Return	Query returns the amplitude in scientific notation
Example	:DDS:AMP 1 /*Set the signal source waveform amplitude to 1V*/ :DDS:AMP? /*The query returns 1.00000e+00*/

13.5 DDS:OFFSet

Command	:DDS:OFFSet <offset> :DDS:OFFSet?
Description	Set or query the offset of the source signal
Parameters	:<offset> ::= unit V
Return	The query returns the offset in scientific notation
Example	:DDS:OFFSet 0.5 /*Set the signal source waveform offset 0.5V*/ :DDS:OFFSet? /*The query returns 5.0000e-01*/

13.6 DDS:DUTY

Command	:DDS:DUTY <duty> :DDS:DUTY?

Description	Set or query the duty cycle of the signal source
Parameters	:<duty> ::= 0-99
Return	The query returns the duty cycle value
Example	:DDS: DUTY 50 /*Set the signal source signal duty cycle to 50%*/ :DDS:DUTY? /*The query returns 50*/

13.7 DDS:WAVE:MODE

Command	:DDS:WAVE:MODE <bool> :DDS:WAVE:MODE?
Description	Set or query the modulation status of the signal source
Parameters	:<bool> ::= {{1 | ON} | {0 | OFF}
Return	The query returns ON, OFF
Example	:DDS:WAVE:MODE ON /*Turn on signal source modulation*/ :DDS:WAVE:MODE? /*The query returns ON*/

13.8 DDS:MODE:TYPE

Command	:DDS:MODE:TYPE <type> :DDS:MODE:TYPE?
Description	Set or query the modulation type of the signal source
Parameters	:<type> ::= { AM| FM}
Return	The query returns AM, FM
Example	:DDS:MODE:TYPE AM /*Set the signal source modulation type to AM*/ :DDS:MODE:TYPE? /*The query returns AM*/

13.9 DDS:MODE:WAVE:TYPE

Command	:DDS:MODE:WAVE:TYPE <type> :DDS:MODE:WAVE:TYPE?
Description	Set or query the modulation wave type when the signal source is modulated
Parameters	:<type> ::= { SINE| SQUAre| RAMP}
Return	The query returns SINE, SQUAre, RAMP
Example	:DDS:MODE:WAVE:TYPE SINE /*Set the signal source modulation wave type to SINE*/ :DDS:MODE:WAVE:TYPE? /*The query returns SINE*/

13.10 DDS:MODE:FREQ

Command	:DDS:MODE:FREQ <freq> :DDS:MODE:FREQ?
Description	Set or query the frequency of the modulating wave when the signal source is modulated
Parameters	:<freq> ::= unit Hz
Return	Query returns frequency in scientific notation
Example	:DDS:MODE:FREQ 1000 /*Set the signal source modulation wave frequency 1khz*/ :DDS:MODE:FREQ? /*The query returns 1.00000e+03*/

13.11 DDS:MODE:DEPThordeviation

Command	:DDS:MODE:DEPThordeviation <value> :DDS:MODE:DEPThordeviation?
Description	Set or query the deviation or depth of signal source modulation
Parameters	The modulation type is AM: <value> ::= modulation depth Modulation type is FM: <value> ::= deviation
Return	When the modulation type is AM, the query returns the value of the modulation depth The query returns deviation when the modulation type is FM
Example	Modulation type is AM :DDS:MODE:DEPThordeviation 50 /*Set the modulation depth to 50*/ :DDS:MODE:DEPThordeviation? /*The query returns 50*/ Modulation type is FM :DDS:MODE:DEPThordeviation 1000 /*Set the deviation to 1khz*/ :DDS:MODE:DEPThordeviation? /*The query returns 1000*/

13.12 DDS:BURSt:SWITch

Command	:DDS:BURSt:SWITch <bool> :DDS:BURSt:SWITch?
Description	Set or query the signal source burst status
Parameters	:<bool> ::= {{1 | ON} | {0 | OFF}
Return	The query returns ON, OFF
Example	:DDS:BURSt:SWITch ON /*Turn on the signal source burst*/ :DDS:BURSt:SWITch? /*The query returns ON*/

13.13 DDS:BURSt:TYPE

Command	:DDS:BURSt:TYPE <type> :DDS:BURSt:TYPE?
Description	Set or query the signal source burst type
Parameters	:<type> ::= { N_CYCLE| INFInit}
Return	The query returns N_CYCLE, INFInit
Example	:DDS:BURSt:TYPE N_CYCLE /*Set the burst type to N_CYCLE*/ :DDS:BURSt:TYPE? /*The query returns N_CYCLE*/

13.14 DDS:BURSt:CNT

Command	:DDS:BURSt:CNT <cnt> :DDS:BURSt:CNT?
Description	Set or query the number of signal cycles of the signal source
Parameters	:<value> ::= integer
Return	The query returns an integer
Example	:DDS:BURSt:CNT 2 /*Set the number of cycles to 2*/ :DDS:BURSt:CNT? /*The query returns 2*/

13.15 DDS:BURSt:TRIGger

Command	:DDS:BURSt:TRIGger

Description	Send a signal source burst once

13.16 DDS:ARB:DAC16:BIN

Command	:DDS:ARB:DAC16:BIN <binary_block_data>
Description	Download waveform data, <binary_block_data> means to download binary data, <binary_block_data> is a binary data block starting with # sign, "#508192" binary data, "5" after "#" means there are 5 characters representing data length information , "8192" means that there are 8192 bytes representing binary data. Each waveform point corresponds to a binary number of two bytes (for example, the point 1024 corresponds to a binary number of 0x0400, the low byte of the data is in front, and the high byte is in the back, so it is 0004), so the number of bytes must be an even number. Note: The number of arbitrary waveform points must be 4096