

Mixed-Domain Oscilloscope

MDO-2000EG & MDO-2000EX

PROGRAMMING MANUAL



ISO-9001 CERTIFIED MANUFACTURER

GW INSTEK

July 2017

This manual contains proprietary information which is protected by copyright. All rights are reserved. No part of this manual may be photocopied, reproduced or translated to another language without prior written consent of Good Will Corporation.

The information in this manual was correct at the time of printing. However, Good Will continues to improve products and reserves the right to change specifications, equipment, and maintenance procedures at any time without notice.

Windows is a registered trademark of Microsoft Corporation in the United States and other countries.

Good Will Instrument Co., Ltd.

No. 7-1, Jhongsing Rd., Tucheng Dist., New Taipei City 236, Taiwan.

Table of Contents

INTERFACE OVERVIEW	3
Interface Configuration	3
COMMAND OVERVIEW	13
Command Syntax	13
List of Commands in Functional Order	15
COMMAND DETAILS	30
Common Commands	32
Acquisition Commands	38
Autoscale Commands	45
Vertical Commands	46
Math Commands	52
Cursor Commands	61
Display Commands	72
Hardcopy Commands	76
Measure Commands	79
Measurement Commands	105
Reference Commands	112
Run Command	115
Stop Command	115
Single Command	115
Force Command	116
Timebase Commands	117
Trigger Commands	120
System Commands	157
Save/Recall Commands	158
Ethernet Command	163
Time Command	163
Bus Decode Commands	164
Mark Commands	178

Search Commands	180
Label Commands	213
Segment Commands	219
DVM Commands.....	228
Go_NoGo Commands	230
AWG Commands.....	236
Data Logging Commands.....	255
Remote Disk Commands.....	258
DMM Commands (For the MDO-2000EX series only)	261
Spectrum Analyzer Commands	267
Power Supply Commands (For the MDO- 2000EX series only).....	281
USB Delay Command.....	284
APPENDIX.....	285
Error messages	285
INDEX.....	289

INTERFACE OVERVIEW

This manual describes how to use the MDO-2000EG/2000EX's remote command functionality and lists the command details. The Overview chapter describes how to configure the USB and Ethernet remote control interfaces.

Interface Configuration

Configure USB Interface

USB Configuration	PC side connector	Type A, host
	MDO-2000EG/2000EX side connector	Type B, device
	Speed	1.1/2.0
	USB Class	CDC (communications device class)

- Panel Operation
1. Press the Utility key.
 2. Press *I/O* from the bottom menu.
 3. Press *USB Device Port* from the side menu and select *Computer*.
 4. Connect the USB cable to the rear panel device port.



- When the PC asks for the USB driver, select the USB driver included on the accompanying User Manual CD or download the driver from the GW Instek website, www.gwinstek.com, in the MDO-2000EG/2000EX Download section. The driver automatically sets the MDO-2000EG/2000EX as a serial COM port (Shown as VPO in the PORTS node).

USB Functionality Check

Terminal Application

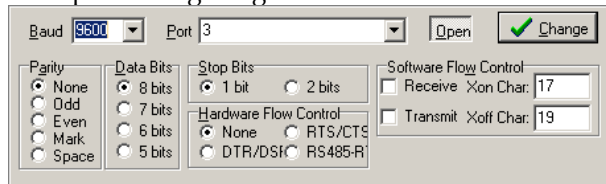
Invoke a terminal application such as RealTerm.

Set the COM port, baud rate, stop bit, data bit, and parity accordingly.

To check the COM port number and associated port settings, see the Device Manager in the PC. For Windows 7:

Control panel → Hardware and Sound → Device Manager

Example: Configuring RealTerm:



Functionality Check

Key in this query command via the terminal application.

`*idn?`

This should return the Manufacturer, Model number, Serial number, and Firmware version in the following format.

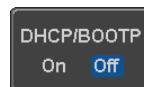
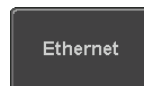
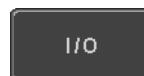
GW,MDO-2202EG,PXXXXXX,V1.00

Configure the Ethernet Interface

Ethernet Configuration	MAC Address	Domain Name
	Instrument Name	DNS IP Address
	User Password	Gateway IP Address
	Instrument IP Address	Subnet Mask

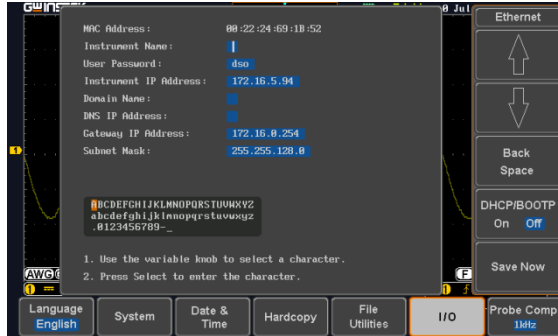
Background The Ethernet interface is used for remote control using a socket server connection. For details, please see the Socket Server section on page 7.

- Panel Operation**
1. Connect the Ethernet cable to the LAN port on the rear panel.
 2. Press the *Utility* key.
 3. Press *I/O* from the bottom menu.
 4. Press *Ethernet* from the side menu.
 5. Set *DHCP/BOOTP* to *On* or *Off* from the side menu.

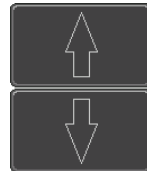


Note

IP addresses will automatically be assigned with DHCP/BOOTP set to on. For Static IP Addresses, DHCP/BOOTP should be set to off.

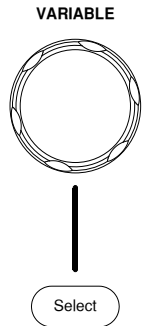


6. Use the *Up* and *Down* arrows on the side menu to navigate to each Ethernet configuration item.



Items MAC Address, Instrument Name, User Password, Instrument IP Address, Domain Name, DNS IP Address, Gateway IP Address, Subnet Mask.

7. Use the *Variable* knob to highlight a character and use the *Select* key to choose a character.



Press *Backspace* to delete a character.



Press *Save Now* to save the configuration. Complete will be displayed when successful.



Configure Socket Server

The MDO-2000EG/2000EX supports socket server functionality for direct two-way communication with a client PC or device over LAN. By default, the Socket Server is off.

Configure Socket Server 1. Configure the IP address for the MDO-2000EG/2000EX. Page 5

2. Press the *Utility* key.



3. Press *I/O* from the bottom menu.



4. Press *Socket Server* from the side menu.



5. Press *Select Port* and choose the port number with the Variable knob.



Range 1024~65535

6. Press *Set Port* to confirm the port number.



7. The *Current Port* icon will update to the new port number.




8. Press *Server* and turn the socket server On.



Socket Server Functionality Check

NI Measurement and Automation Explorer To test the socket server functionality, National Instruments Measurement and Automation Explorer can be used. This program is available on the NI website, www.ni.com.

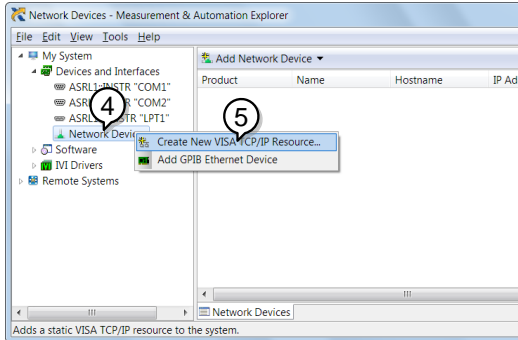
- | | | |
|-----------|--|---|
| Operation | 1. Configure the IP address for the MDO-2000EG/2000EX. | Page 5 |
| | 2. Configure the socket port. | Page 7 |
| | 3. Start the NI Measurement and Automation Explorer (MAX) program. Using Windows, press: |  |

Start>All Programs>National Instruments>Measurement & Automation

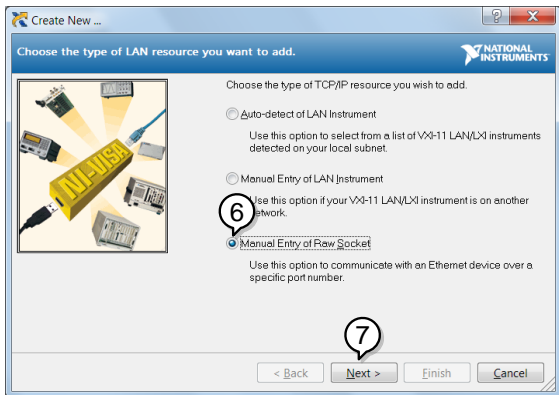


4. From the Configuration panel access;

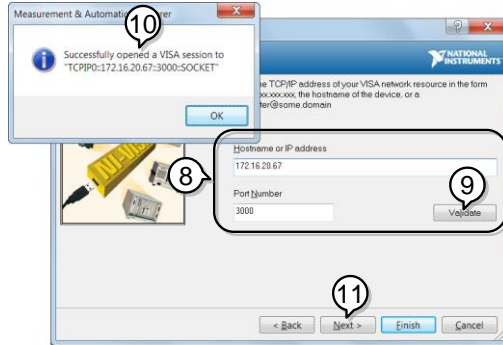
My System>Devices and Interfaces>Network Devices
5. Right click *Network Devices* and select *Create New Visa TCP/IP Resource...*



6. Select *Manual Entry of Raw Socket* from the popup window.
7. Click *Next*.



8. Enter the MDO-2000EG/2000EX's IP address and socket port number.
9. Click *Validate*.
10. A popup will appear to tell you if a VISA socket session was successfully created.
11. Click *Next*.



12. Choose an alias for the socket connection if you like.

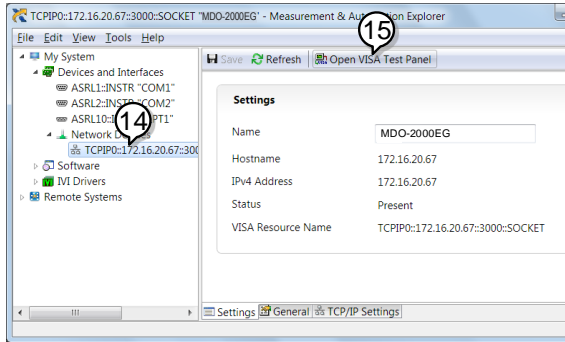
13. Click *Finish* to finish the configuration.



14. The MDO-2000EG/2000EX will now appear under Network Devices in the Configuration Panel.

Functionality Check

15. Click the *Open Visa Test Panel* to send a remote command to the MDO-2000EG/2000EX.

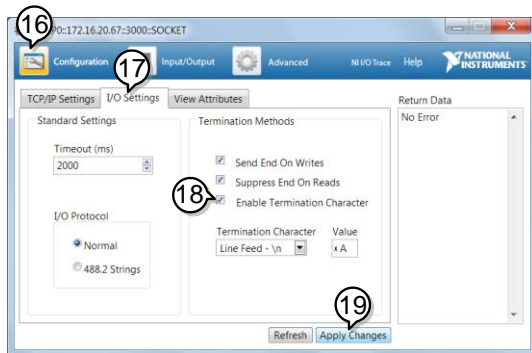


16. Click on the *Configuration* icon.

17. Select the *I/O Settings* tab.

18. Mark the *Enable Termination Character* checkbox. Make sure the termination character is a line feed (/n, value: xA).

19. Click *Apply Changes*.



20. Click the *Input/Output* icon.
21. Make sure **IDN?* query is selected in the *Select or Enter Command* drop box.
22. Click on *Query*.
23. The manufacturer, model number, serial number and firmware version will be displayed in the buffer. For example:
GW,MDO-2202EG,PXXXXXX,V1.00



C COMMAND OVERVIEW

The Command overview chapter lists all MDO-2000EG/2000EX commands in functional order as well as alphabetical order. The command syntax section shows you the basic syntax rules you have to apply when using commands.

Command Syntax

Compatible standard

- USB CDC_ACM compatible
- SCPI, 1994 (partially compatible)

Command forms

Commands and queries have two different forms, long and short. The command syntax is written with the short form of the command in capitals and the remainder (long form) in lower case.

```

      Long      Long
      |         |
  {-----}  {-----}
  |         |
:TIMEbase:SCALE?
  |         |
  Short    Short
  
```

The commands can be written in capitals or lower-case, just so long as the short or long forms are complete. An incomplete command will not be recognized.

Below are examples of correctly written commands.

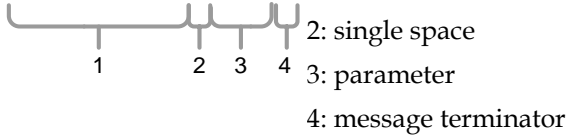
```

LONG  :TIMEbase:SCALE?  :TIMEBASE:SCALE?
      :timebase:scale?
  
```

```

SHORT :TIM:SCAL?      :TIM:SCAL?
  
```

Command format :TIMEbase:SCALE<NR3>LF 1: command header



Parameter	Type	Description	Example
	<Boolean>	boolean logic	0, 1
	<NR1>	Integers	0, 1, 2, 3
	<NR2>	floating point	0.1, 3.14, 8.5
	<NR3>	floating point with an exponent	4.5e-1, 8.25e+1
	<NRf>	any of NR1, 2, 3	1, 1.5, 4.5e-1

Message terminator LF line feed code

Note Commands are non-case sensitive.

List of Commands in Functional Order

Common	*IDN?	32
	*LRN?	32
	*SAV	33
	*RCL	33
	*RST	34
	*CLS	34
	*ESE	34
	*ESR	35
	*OPC	35
	*SRE	36
	*STB	36
Acquisition	:ACQuire:AVERAge	38
	:ACQuire:MODE	39
	:ACQuire<X>:MEMory?	39
	:ACQuire:FILTer:SOURce	41
	:ACQuire:FILTer	42
	:ACQuire:FILTer:FREQuency	42
	:ACQuire:FILTer:TRACking	42
	:ACQuire<X>:STATe?	43
	:ACQuire:INTERpolation	43
	:ACQuire:RECOrdlength	43
:HEADer	44	
Autoscale	:AUTOSet	45
	:AUTORSET:MODE	45
Vertical Scale	:CHANnel<X>:BWLimit	46
	:CHANnel<X>:COUPling	47
	:CHANnel<X>:DESKew	47
	:CHANnel<X>:DISPlay	47
	:CHANnel<X>:EXPanD	48
	:CHANnel<X>:IMPedance?	48

	:CHANnel<X>:INVert	49
	:CHANnel<X>:POSition	49
	:CHANnel<X>:PROBe:RATio	50
	:CHANnel<X>:PROBe:TYPe	50
	:CHANnel<X>:SCALe	50
Math	:MATH:DISP	52
	:MATH:TYPe	53
	:MATH:DUAL:SOURce<X>	53
	:MATH:DUAL:OPERator	53
	:MATH:DUAL:POSition	54
	:MATH:DUAL:SCALe	54
	:MATH:FFT:SOURce	55
	:MATH:FFT:MAG	55
	:MATH:FFT:WINDow	55
	:MATH:FFT:POSition	56
	:MATH:FFT:SCALe	56
	:MATH:FFT:HORizontal:SCALe	57
	:MATH:FFT:HORizontal:POSition	57
	:MATH:DEFine	57
	:MATHVAR?	58
	:MATHVAR:VAR<X>	59
	:MATH:ADVanced:POSition	59
	:MATH:ADVanced:SCALe	59
Cursor	:CURSor:MODE	62
	:CURSor:SOURce.....	62
	:CURSor:HUNI	63
	:CURSor:HUSE	63
	:CURSor:VUNI	64
	:CURSor:VUSE	64
	:CURSor:DDT.....	64
	:CURSor:H1Position	65
	:CURSor:H2Position.....	65
	:CURSor:HDELta	66
	:CURSor:V1Position.....	66

	:CURSor:V2Position.....	66
	:CURSor:VDELta	67
	:CURSor:XY:RECTangular:X:POSition<X>	67
	:CURSor:XY:RECTangular:X:DELta	67
	:CURSor:XY:RECTangular:Y:POSition<X>	68
	:CURSor:XY:RECTangular:Y:DELta	68
	:CURSor:XY:POLar:RADIUS:POSition<X>.....	68
	:CURSor:XY:POLar:RADIUS:DELta	69
	:CURSor:XY:POLar:THETA:POSition<X>	69
	:CURSor:XY:POLar:THETA:DELta	69
	:CURSor:XY:PRODuct:POSition<X>	70
	:CURSor:XY:PRODuct:DELta.....	70
	:CURSor:XY:RATio:POSition<X>.....	70
	:CURSor:XY:RATio:DELta	71
<hr/>		
Display	:DISPlay:INTensity:WAVEform.....	72
	:DISPlay:INTensity:GRATICule.....	72
	:DISPlay:INTensity:BACKLight	73
	:DISPlay:INTensity:BACKLight:AUTODim:ENABle	73
	:DISPlay:INTENSITy:BACKLight:AUTODim:TIME	73
	:DISPlay:PERSistence	74
	:DISPlay:GRATICule	74
	:DISPlay:WAVEform.....	75
	:DISPlay:OUTPut	75
<hr/>		
Hardcopy	:HARDcopy:START	76
	:HARDcopy:MODE	76
	:HARDcopy:PRINTINKSaver	77
	:HARDcopy:SAVEINKSaver	77
	:HARDcopy:SAVEFORMat	77
	:HARDcopy:ASSIGN.....	78
<hr/>		
Measure	:MEASure:GATing.....	80
	:MEASure:SOURce	80
	:MEASure:METHod	81
	:MEASurement:REFLevel:PERCent:HIGH.....	81

:MEASUrement:REFLevel:PERCent:LOW	82
:MEASUrement:REFLevel:PERCent:MID	82
:MEASUrement:REFLevel:PERCent:MID2	82
:MEASUre:FALL.....	83
:MEASUre:FOVShoot	83
:MEASUre:FPReshoot	84
:MEASUre:FREQuency	84
:MEASUre:NWIDth	85
:MEASUre:PDUTy	85
:MEASUre:PERiod	86
:MEASUre:PWIDth	86
:MEASUre:RISe	87
:MEASUre:ROVShoot.....	87
:MEASUre:RPPReshoot.....	88
:MEASUre:PPULSE.....	89
:MEASUre:NPULSE.....	89
:MEASUre:PEDGE	90
:MEASUre:NEDGE	90
:MEASUre:AMPlitude.....	91
:MEASUre:MEAN	91
:MEASUre:CMean	92
:MEASUre:HIGH	93
:MEASUre:LOW.....	93
:MEASUre:MAX.....	94
:MEASUre:MIN	94
:MEASUre:PK2PK.....	95
:MEASUre:RMS	95
:MEASUre:CRMS.....	96
:MEASUre:AREa	97
:MEASUre:CARea	97
:MEASUre:FRRDelay	98
:MEASUre:FRFDelay	98
:MEASUre:FFRDelay	99
:MEASUre:FFFDelay.....	100
:MEASUre:FLI.....	100

	:MEASure:LRRDelay	101
	:MEASure:LRFDelay.....	101
	:MEASure:LFRDelay.....	102
	:MEASure:LFFDelay.....	103
	:MEASure:PFLI.....	103
	:MEASure:PHAsE.....	104
Measurement	:MEASUrement:MEAS<X>:SOURCE<X>	105
	:MEASUrement:MEAS<X>:TYPe.....	106
	:MEASUrement:MEAS<X>:STATe	106
	:MEASUrement:MEAS<X>:VALue.....	107
	:MEASUrement:MEAS<X>:MAXimum	108
	:MEASUrement:MEAS<X>:MEAN	109
	:MEASUrement:MEAS<X>:MINImum	109
	:MEASUrement:MEAS<X>:STDdev	110
	:MEASUrement:STATIstics:MODE	111
	:MEASUrement:STATIstics:WEIghting	111
	:MEASUrement:STATIstics	111
Reference	:REF<X>:DISPlay.....	112
	:REF<X>:TIMebase:POSition	112
	:REF<X>:TIMebase:SCALE	113
	:REF<X>:OFFSet	113
	:REF<x>:SCALE	114
Run	:RUN	115
Stop	:STOP.....	115
Single	:SINGLe.....	115
Force	:FORCe.....	116
Timebase	:TIMebase:EXPand	117
	:TIMebase:POSition	117

	:TIMebase:SCALe	118
	:TIMebase:MODE	118
	:TIMebase:WINDow:POSition	118
	:TIMebase:WINDow:SCALe	119
Trigger	:TRIGger:FREQuency.....	122
	:TRIGger:TYPe	122
	:TRIGger:SOURce	123
	:TRIGger:COUPLe.....	123
	:TRIGger:NREJ	123
	:TRIGger:MODE.....	124
	:TRIGger:HOLDoff.....	124
	:TRIGger:LEVel.....	124
	:TRIGger:HLEVel	125
	:TRIGger:LLEVel.....	125
	:TRIGger:EDGE:SLOP.....	126
	:TRIGger:DELay:SLOP	126
	:TRIGger:DELay:TYPe	127
	:TRIGger:DELay:TIMe.....	127
	:TRIGger:DELay:EVENT.....	127
	:TRIGger:DELay:LEVel	128
	:TRIGger:PULSEwidth:POLarity	128
	:TRIGger:RUNT:POLarity	129
	:TRIGger:RUNT:WHEn.....	129
	:TRIGger:RUNT:TIMe.....	130
	:TRIGger:RISEFall:SLOP.....	130
	:TRIGger:RISEFall:WHEn	130
	:TRIGger:RISEFall:TIMe	131
	:TRIGger:VIDeo:TYPe	131
	:TRIGger:VIDeo:FIELd	132
	:TRIGger:VIDeo:LINE	132
	:TRIGger:VIDeo:POLarity	133
	:TRIGger:PULSE:WHEn	133
	:TRIGger:PULSE:TIMe	134
	:TRIGger:TIMEOut:WHEn.....	134

:TRIGger:TIMEOut:TIMER	135
:TRIGger:ALTErnate	135
:TRIGger:STATe	135
:TRIGger:EXTErnal:PROBe:TYPe.....	136
:TRIGger:EXTErnal:PROBe:RATio	137
:TRIGger:BUS:TYPe	137
:TRIGger:BUS:THReshold:CH<x>	137
:TRIGger:BUS:B1:I2C:CONDition	138
:TRIGger:BUS:B1:I2C:ADDReSS:MODE.....	139
:TRIGger:BUS:B1:I2C:ADDReSS:TYPe.....	139
:TRIGger:BUS:B1:I2C:ADDReSS:VALue	140
:TRIGger:BUS:B1:I2C:ADDReSS:DIRectioN.....	140
:TRIGger:BUS:B1:I2C:DATA:SIZE	141
:TRIGger:BUS:B1:I2C:DATA:VALue	141
:TRIGger:BUS:B1:UART:CONDition	142
:TRIGger:BUS:B1:UART:RX:DATA:SIZE	143
:TRIGger:BUS:B1:UART:RX:DATA:VALue	143
:TRIGger:BUS:B1:UART:TX:DATA:SIZE.....	144
:TRIGger:BUS:B1:UART:TX:DATA:VALue	144
:TRIGger:BUS:B1:SPI:CONDition	145
:TRIGger:BUS:B1:SPI:DATA:SIZE	146
:TRIGger:BUS:B1:SPI:DATA:MISO:VALue	146
:TRIGger:BUS:B1:SPI:DATA:MOSI:VALue	147
:TRIGger:BUS:B1:CAN:CONDition	148
:TRIGger:BUS:B1:CAN:FRAMeType	148
:TRIGger:BUS:B1:CAN:IDentifier:MODE.....	149
:TRIGger:BUS:B1:CAN:IDentifier:VALue	149
:TRIGger:BUS:B1:CAN:IDentifier:DIRectioN.....	150
:TRIGger:BUS:B1:CAN:DATA:QUALifier	150
:TRIGger:BUS:B1:CAN:DATA:SIZE.....	151
:TRIGger:BUS:B1:CAN:DATA:VALue	152
:TRIGger:BUS:B1:LIN:CONDition	152
:TRIGger:BUS:B1:LIN:DATA:QUALifier.....	153
:TRIGger:BUS:B1:LIN:DATA:SIZE	154
:TRIGger:BUS:B1:LIN:DATA:VALue	154

	:TRIGger:BUS:B1:LIN:ERRType.....	155
	:TRIGger:BUS:B1:LIN:IDentifier:VALue	156
<hr/>		
System	:SYSTem:LOCK	157
commands	:SYSTem:ERRor	157
<hr/>		
Save/Recall	:RECALL:SETUp	158
	:RECALL:WAVEform	158
	:SAVE:IMAGe	159
	:SAVE:IMAGe:FILEFormat.....	159
	:SAVE:IMAGe:INKSaver.....	160
	:SAVE:SETUp	160
	:SAVE:WAVEform.....	161
	:SAVE:WAVEform:FILEFormat	162
<hr/>		
Ethernet	:ETHERnet:DHCP	163
Command		
<hr/>		
Time Command	:DATE	163
<hr/>		
Bus Decode	:BUS1	165
Commands	:BUS1:STATE	165
	:BUS1:TYPE.....	165
	:BUS1:INPut	166
	:BUS1:I2C:ADDRes:RWINClude	166
	:BUS1:I2C:SCLK:SOURce	167
	:BUS1:I2C:SDA:SOURce	167
	:BUS1:UART:BITRate.....	167
	:BUS1:UART:DATABits.....	168
	:BUS1:UART:PARItY.....	168
	:BUS1:UART:PACKet	168
	:BUS1:UART:EOFPAcket	169
	:BUS1:UART:TX:SOURce	169
	:BUS1:UART:RX:SOURce	169
	:BUS1:SPI:SCLK:POLARity	170
	:BUS1:SPI:SS:POLARity.....	170
	:BUS1:SPI:WORDSize	170

	:BUS1:SPI:BITOrder	171
	:BUS1:SPI:SCLK:SOURce	171
	:BUS1:SPI:SS:SOURce	171
	:BUS1:SPI:MOSI:SOURce	172
	:BUS1:SPI:MISO:SOURce	172
	:BUS1:DISplay:FORMAt	172
	:LISTer:DATA	173
	:BUS1:CAN:SOURce	173
	:BUS1:CAN:PROBe	173
	:BUS1:CAN:SAMPLEpoint	174
	:BUS1:CAN:BITRate	174
	:BUS1:LIN:BITRate	175
	:BUS1:LIN:IDFORmat	175
	:BUS1:LIN:POLARity	175
	:BUS1:LIN:SAMPLEpoint	176
	:BUS1:LIN:SOURce	176
	:BUS1:LIN:STANDard	176

Mark Commands	:MARK	178
	:MARK:CREATe	178
	:MARK:DELEte	179

Search	:SEARCH:COpy	181
Commands	:SEARCH:STATe	182
	:SEARCH:TOTAL	182
	:SEARCH:TRIGger:TYPe	182
	:SEARCH:TRIGger:SOURce	183
	:SEARCH:TRIGger:EDGE:SLOP	183
	:SEARCH:TRIGger:LEVel	184
	:SEARCH:TRIGger:HLEVel	184
	:SEARCH:TRIGger:LLEVel	185
	:SEARCH:TRIGger:PULSEWidth:POLarity	185
	:SEARCH:TRIGger:RUNT:POLarity	186
	:SEARCH:TRIGger:RISEFall:SLOP	186
	:SEARCH:TRIGger:PULSe:WHEn	187

:SEARCH:TRIGger:PULSe:TIME.....	187
:SEARCH:TRIGger:RUNT:WHEn	188
:SEARCH:TRIGger:RUNT:TIME	188
:SEARCH:TRIGger:RISEFall:WHEn	189
:SEARCH:TRIGger:RISEFall:TIME	189
:SEARCH:TRIGger:BUS:TYPe.....	190
:SEARCH:TRIGger:BUS:B1:I2C:CONDition	190
:SEARCH:TRIGger:BUS:B1:I2C:ADDReSS:MODE	191
:SEARCH:TRIGger:BUS:B1:I2C:ADDReSS:TYPe	191
:SEARCH:TRIGger:BUS:B1:I2C:ADDReSS:VALue.....	192
:SEARCH:TRIGger:BUS:B1:I2C:ADDReSS	
:DIRection	193
:SEARCH:TRIGger:BUS:B1:I2C:DATA:SIZE	193
:SEARCH:TRIGger:BUS:B1:I2C:DATA:VALue	194
:SEARCH:TRIGger:BUS:B1:UART:CONDition.....	195
:SEARCH:TRIGger:BUS:B1:UART:RX:DATA:SIZE.....	196
:SEARCH:TRIGger:BUS:B1:UART:RX:DATA:VALue..	196
:SEARCH:TRIGger:BUS:B1:UART:TX:DATA:SIZE	197
:SEARCH:TRIGger:BUS:B1:UART:TX:DATA:VALue..	197
:SEARCH:TRIGger:BUS:B1:SPI:CONDition.....	198
:SEARCH:TRIGger:BUS:B1:SPI:DATA:SIZE.....	199
:SEARCH:TRIGger:BUS:B1:SPI:DATA:MISO:VALue.	199
:SEARCH:TRIGger:BUS:B1:SPI:DATA:MOSI:VALue.	200
:SEARCH:TRIGger:BUS:B1:CAN:CONDition	201
:SEARCH:TRIGger:BUS:B1:CAN:FRAMeType	202
SEARCH:TRIGger:BUS:B1:CAN:IDentifier:MODE ...	202
:SEARCH:TRIGger:BUS:B1:CAN:IDentifier:VALue...	203
:SEARCH:TRIGger:BUS:B1:CAN:IDentifier	
:DIRection	203
:SEARCH:TRIGger:BUS:B1:CAN:DATA:QUALifier ...	204
:SEARCH:TRIGger:BUS:B1:CAN:DATA:SIZE	205
:SEARCH:TRIGger:BUS:B1:CAN:DATA:VALue	205
:SEARCH:TRIGger:BUS:B1:LIN:CONDition	206
:SEARCH:TRIGger:BUS:B1:LIN:DATA:QUALifier	207
:SEARCH:TRIGger:BUS:B1:LIN:DATA:SIZE	208

	:SEARCH:TRIGger:BUS:B1:LIN:DATa:VALue	208
	:SEARCH:TRIGger:BUS:B1:LIN:ERRTYPE	209
	:SEARCH:TRIGger:BUS:B1:LIN:IDentifier:VALue.....	209
	:SEARCH:FFTPeak:METHod	210
	:SEARCH:FFTPeak:METHod:MPEak.....	211
	:SEARCH:FFTPeak:SINFo.....	211
	:SEARCH:FFTPeak:LIST	212
<hr/>		
Label Commands	:CHANnel<X>:LABel.....	213
	:CHANnel<X>:LABel:DISPlay	214
	:REF<X>:LABel	214
	:REF<X>:LABel:DISPlay	215
	:BUS1:LABel	216
	:BUS1:LABel:DISPlay	216
	:SET<X>:LABel	217
<hr/>		
Segment	:SEGMENTS:STATE	219
Commands	:SEGMENTS:CURRent	220
	:SEGMENTS:TOTALnum	220
	:SEGMENTS:TIME	221
	:SEGMENTS:DISPALL	221
	:SEGMENTS:MEASure:MODE	221
	:SEGMENTS:MEASure:PLOT:SOURce	222
	:SEGMENTS:MEASure:PLOT:DIVide	222
	:SEGMENTS:MEASure:PLOT:SElect	223
	:SEGMENTS:MEASure:PLOT:RESults.....	223
	:SEGMENTS:MEASure:TABLE:SOURce	224
	:SEGMENTS:MEASure:TABLE:SElect.....	224
	:SEGMENTS:MEASure:TABLE:LIST.....	225
	:SEGMENTS:MEASure:TABLE:SAVE	225
	:SEGMENTS:SAVE	225
	:SEGMENTS:SAVE:SOURce	226
	:SEGMENTS:SAVE:SElect:START	226
	:SEGMENTS:SAVE:SElect:END	227

DVM Commands	:DVM:STATE	228
	:DVM:SOURce	228
	:DVM:MODE.....	229
	:DVM:VALue	229
<hr/>		
Go-NoGo	:GONogo:CLear	230
Commands	:GONogo:EXECute	230
	:GONogo:FUNctIon.....	231
	:GONogo:NGCount	231
	:GONogo:NGDefine.....	231
	:GONogo:SOURce	232
	:GONogo:VIOLation	232
	:GONogo:SCRipt	232
	:TEMPlate:MODE	233
	:TEMPlate:MAXimum.....	233
	:TEMPlate:MINimum	233
	:TEMPlate:POSition:MAXimum.....	234
	:TEMPlate:POSition:MINimum	234
	:TEMPlate:SAVe:MAXimum	234
	:TEMPlate:SAVe:MINimum	235
	:TEMPlate:TOLerance.....	235
	:TEMPlate:SAVe:AUTo	235
<hr/>		
AWG Commands	:AWG<x>:AMPlitude.....	237
	:AWG<x>:FREQuency.....	237
	:AWG<x>:FUNctIon	238
	:AWG<x>:OFFSet	239
	:AWG<x>:OUTPut:LOAd:IMPEDance	239
	:AWG<x>:OUTPut:STATE	239
	:AWG<x>:PHAsE.....	240
	:AWG<x>:PULSe:DUTYcycle	240
	:AWG<x>:RAMP:SYMmetry	240
	:AWG<x>:MODulation:STATE	241
	:AWG<x>:MODulation:TYPe.....	241
	:AWG<x>:MODulation:AM:DEPth	241

:AWG<x>:MODulation:AM:FREQ	242
:AWG<x>:MODulation:AM:SHApe	242
:AWG<x>:MODulation:AM:PHase	242
:AWG<x>:MODulation:AM:DUTYcycle.....	243
:AWG<x>:MODulation:AM:SYMmetry	243
:AWG<x>:MODulation:AM:RATE	244
:AWG<x>:MODulation:FM:DEV	244
:AWG<x>:MODulation:FM:FREQ	245
:AWG<x>:MODulation:FM:SHApe	245
:AWG<x>:MODulation:FM:PHase	245
:AWG<x>:MODulation:FM:DUTYcycle.....	246
:AWG<x>:MODulation:FM:SYMmetry.....	246
:AWG<x>:MODulation:FM:RATE.....	246
:AWG<x>:MODulation:FSK:FREQ	247
:AWG<x>:MODulation:FSK:RATE	247
:AWG<x>:SWEep:STATE	248
:AWG<x>:SWEep:TYPe	248
:AWG<x>:SWEep:START	248
:AWG<x>:SWEep:STOP	249
:AWG<x>:SWEep:TIME.....	249
:AWG<x>:SWEep:SPAN.....	249
:AWG<x>:SWEep:CENTer	250
:AWG<x>:ARBitrary:EDIT:NUMPOINT	250
:AWG<x>:ARBitrary:EDIT:FUNCTion.....	250
:AWG<x>:ARBitrary:SAVe:WAVEform	251
:AWG<x>:ARBitrary:LOAd:WAVEform	251
:AWG<x>:ARBitrary:EDIT:COpy.....	252
:AWG<x>:ARBitrary:EDIT:CLear.....	252
:AWG<x>:ARBitrary:EDIT:LINE	253
:AWG<x>:ARBitrary:EDIT:SCALE.....	253
:AWG<x>:ARBitrary:EDIT:POINT.....	253
:AWG<x>:ARBitrary:EDIT:POINT:ADD.....	254
:AWG<x>:ARBitrary:EDIT:POINT:DELEte.....	254

Data Logging	:DATALOG:STATE	255	
Commands	:DATALOG:SOURce	255	
	:DATALOG:SAVe	256	
	:DATALOG:INTerval	256	
	:DATALOG:DURation.....	257	
	<hr/>		
Remote Disk	:REMOTEDisk:IPADdress	258	
	Commands	:REMOTEDisk:PATHName	258
		:REMOTEDisk:USERName	259
		:REMOTEDisk:PASSWOrd	259
		:REMOTEDisk:MOUNT	259
		:REMOTEDisk:AUTOMount	260
<hr/>			
DMM	:DMM.....	261	
	Commands (For the MDO-2000EX series only)	:DMM:STATE.....	261
		:DMM:VALue	262
		:DMM:HOLD	262
		:DMM:MMIN	262
		:DMM:MODE	263
		:DMM:MODE:RANGe	264
		:DMM:TEMPerature:UNITs	264
		:DMM:TEMPerature:TYPe.....	265
		:DMM:TEMPerature:SIM	265
<hr/>			
Spectrum Analyzer Commands	:SA:MEMory?	267	
	:SA:MEMory:SOURce.....	270	
	:SA:STATE	270	
	:SA:SOURce	270	
	:SA:SPECTRUMTrace	271	
	:SElect:NORMal.....	271	
	:SElect:MAXHold	271	
	:SElect:MINHold	272	
	:SElect:AVErage	272	
	:SA:AVErage:NUMAVg	273	
	:SA:DETECTionmethod:MODE.....	273	
	<hr/>		

	:SA:DETECTIonmethod:MAXHold	273
	:SA:DETECTIonmethod:MINHold	274
	:SA:DETECTIonmethod:NORMAl.....	274
	:SA:DETECTIonmethod:AVErage	275
	:SA:FREQuency	275
	:SA:SPAN	276
	:SA:START	276
	:SA:STOP.....	276
	:SA:RBW:MODE.....	277
	:SA:RBW	277
	:SA:SPANRbwratio	278
	:SA:WINDow	278
	:SA:UNIts	279
	:SA:SCALE	279
	:SA:POSItion	280

Power Supply	:POWERSupply:OUTPut<X>	281
Commands (For	:POWERSupply:OUTPut<X>:VOLTage	281
the MDO-2000EX	:POWERSupply:OUTPut<X>:RECONFigure.....	282
series only)	:POWERSupply:OUTPut<X>:OCP.....	282
	:POWERSupply:CONFigure	283

USB Delay	:USBDelay	284
Command		

C COMMAND DETAILS

The Command details chapter shows the detailed syntax, equivalent panel operation, and example for each command. For the list of all commands, see page 15.

Common Commands	32
Acquisition Commands	38
Autoscale Commands	45
Vertical Commands	46
Math Commands	52
Cursor Commands	61
Display Commands	72
Hardcopy Commands.....	76
Measure Commands	79
Measurement Commands	105
Reference Commands	112
Run Command	115
Stop Command	115
Single Command.....	115
Force Command.....	116
Timebase Commands.....	117
Trigger Commands.....	120
System Commands	157
Save/Recall Commands.....	158
Ethernet Command	163
Time Command	163
Bus Decode Commands	164
Mark Commands.....	178
Search Commands	180

Label Commands	213
Segment Commands	219
DVM Commands	228
Go_NoGo Commands	230
AWG Commands	236
Data Logging Commands	255
Remote Disk Commands	258
DMM Commands (For the MDO-2000EX series only)	261
Spectrum Analyzer Commands	267
Power Supply Commands (For the MDO- 2000EX series only)	281
USB Delay Command	284

Common Commands

*IDN?	32
*LRN?	32
*SAV	33
*RCL	33
*RST	34
*CLS	34
*ESE	34
*ESR	35
*OPC	35
*SRE	36
*STB	36

*IDN?

→ Query

Description	Returns the manufacturer, model, serial number and version number of the unit.
Syntax	*IDN?
Example	*IDN? GW,MDO-2074E,PXXXXXX,V1.XX

*LRN?

→ Query

Description	Returns the oscilloscope settings as a data string.
Syntax	*LRN?
Example	*LRN? :DISPlay:WAVEform VECTOR;PERSistence 2.400E-01; INTensity:WAVEform 50;INTensity:GRATicule 50;GRATicule FULL;:CHANnel CH1:DISPlay ON;BWLimit FULL;COUPling DC;INVert OFF;POSition -1.600E+00;PROBe:RATio

1.000e+01;PROBe:TYPe VOLTAGE;SCALe 2.000E+01;IMPedance 1E+6;EXPand GROUND;;CHANnel CH2:DISPlay ON;BWLimit FULL;COUPLing DC;INVert OFF;POSition 0.000E+00;PROBe:RATio 1.000e+01;PROBe:TYPe VOLTAGE;SCALe 2.000E+00;IMPedance 1E+6;EXPand GROUND;;CHANnel CH3:DISPlay OFF;BWLimit FULL;COUPLing DC;INVert OFF;POSition 0.000E+00;PROBe:RATio 1.000e+01;PROBe:TYPe VOLTAGE;SCALe 1.000E+00;IMPedance 1E+6;EXPand GROUND;;CHANnel CH4:DISPlay OFF;BWLimit FULL;COUPLing DC;INVert OFF;POSition 0.000E+00;PROBe:RATio 1.000e+01;PROBe:TYPe VOLTAGE;SCALe 1.000E+00;IMPedance 1E+6;EXPand GROUND;;MATH:TYPe FFT;DISP OFF;DUAL:SOURce1 CH1;SOURce2 CH2;OPERator MUL;POSition 0.000E+00;SCALe ?;FFT:SOURce CH1;MAG DB;WINDow HANNING;POSition 2.800E-01;SCALe 2.000E+01;MATH:ADVanced:OPERator DIFF;ADVanced:SOURce CH1;ADVanced:EDIT: SOURce1 CH1;ADVanced:EDIT:S

***SAV**



Description Saves the current panel settings to the selected memory number (setup 1 ~ 20).

Syntax *SAV {1 | 2 | 3 |... | 20}

Example *SAV 1
Saves the current panel settings to Set 1.

***RCL**



Description Recalls a set of panel settings.

Syntax *RCL {1 | 2 | 3 |... | 20}

Example *RCL 1
Recalls the selected setup from Set 1.

***RST**

Set →

Description Resets the MDO-2000EG/2000EX (recalls the default panel settings).

Syntax *RST

***CLS**

Set →

Description Clears the error queue.

Syntax *CLS

***ESE**

Set →

→ Query

Description Sets or queries the Standard Event Status Enable register.

Syntax *ESE <NR1>

Query Syntax *ESE?

Return parameter <NR1> 0~255

Bit Weight	Bit#	Weight	Event	Description
	0	1	OPC	Operation Complete Bit
	1	2	RQC	Not used
	2	4	QYE	Query Error
	3	8	DDE	Device Error
	4	16	EXE	Execution Error
	5	32	CME	Command Error
	6	64	URQ	User Request
	7	128	PON	Power On

Example *ESE?

>4

Indicates that there is a query error.

***ESR**

→ Query

Description Queries the Standard Event Status (Event) register. The Event Status register is cleared after it is read.

Query Syntax *ESR?

Return parameter <NR1> 0~255

Bit Weight	Bit#	Weight	Event	Description
	0	1	OPC	Operation Complete Bit
	1	2	RQC	Not used
	2	4	QYE	Query Error
	3	8	DDE	Device Error
	4	16	EXE	Execution Error
	5	32	CME	Command Error
	6	64	URQ	User Request
	7	128	PON	Power On

Example *ESR?
 >4
 Indicates that there is a query error.

Set →

***OPC**

→ Query

Description The *OPC command sets the OPC bit (bit0) of the Standard Event Status Register when all current commands have been processed.
 The *OPC? Query returns 1 when all the outstanding commands have completed.

Syntax *OPC

Query Syntax *OPC?

Return parameter 1 Returns 1 when all the outstanding commands have completed.

Set →

→ Query

***SRE**

Description Sets or queries the Service Request Enable register. The Service Request Enable register determines which registers of the Status Byte register are able to generate service requests.

Syntax *SRE <NR1>

Query Syntax *SRE?

Parameter/ Return parameter <NR1> 0~255

Bit Weight	Bit#	Weight	Event	Description
	0	1		Not used
	1	2		Not used
	2	4		Not used
	3	8		Not used
	4	16	MAV	Message Available Bit
	5	32	ESB	Event Status Bit
	6	64	MSS	Master Summary Bit
	6	64	RQS	Request Service Bit
	7	128		Not used

Example *SRE?
>48
Indicates that the MAVB and ESB bit are both set.

***STB**

→ Query

Description Queries the bit sum of the Status Byte register with MSS (Master summary Status) replacing the RQS bit (bit 6).

Query Syntax *STB?

Return parameter <NR1> 0 ~ 255

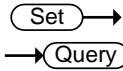
Bit Weight	Bit#	Weight	Event	Description
	0	1		Not used
	1	2		Not used
	2	4		Not used
	3	8		Not used
	4	16	MAV	Message Available Bit
	5	32	ESB	Event Status Bit
	6	64	MSS	Master Summary Bit
	6	64	RQS	Request Service Bit
	7	128		Not used

Example *STB?
>16
Indicates that the MAV bit is set.

Acquisition Commands

:ACQuire:AVERage	38
:ACQuire:MODE	39
:ACQuire<X>:MEMory?	39
:ACQuire:FILTer:SOURce	41
:ACQuire:FILTer	42
:ACQuire:FILTer:FREQuency.....	42
:ACQuire:FILTer:TRACking.....	42
:ACQuire<X>:STATe?	43
:ACQuire:INTERpolation	43
:ACQuire:RECOrdlength.....	43
:HEADer	44

:ACQuire:AVERage



Description	Selects or returns the number of waveform acquisitions that are averaged in the average acquisition mode.
Syntax	:ACQuire:AVERage {<NR1> ?}
Related Commands	:ACQuire:MODE
Parameter	<NR1> 2, 4, 8, 16, 32, 64, 128, 256
Note	Before using this command, select the average acquisition mode. See the example below.
Example	:ACQuire:MODE AVERage :ACQuire:AVERage 2 Selects the average acquisition mode, and sets the average number to 2.

Set →
 → Query

:ACquire:MODE

Description	Selects or returns the acquisition mode.						
Syntax	:ACquire:MODE {SAMPLE PDETECT AVERage ?}						
Related Commands	:ACquire:AVERage						
Parameter	<table style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 15%; border-right: 1px solid black;">SAMPLE</td> <td>Sample mode sampling</td> </tr> <tr> <td style="border-right: 1px solid black;">PDETECT</td> <td>Peak detect sampling</td> </tr> <tr> <td style="border-right: 1px solid black;">AVERage</td> <td>Average sampling mode</td> </tr> </table>	SAMPLE	Sample mode sampling	PDETECT	Peak detect sampling	AVERage	Average sampling mode
SAMPLE	Sample mode sampling						
PDETECT	Peak detect sampling						
AVERage	Average sampling mode						
Example	:ACquire:MODE PDETECT Sets the sampling mode to peak detection.						

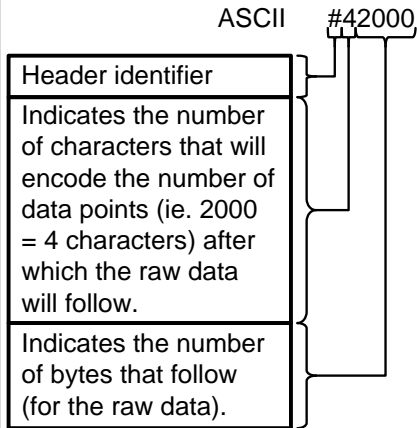
→ Query

:ACquire<X>:MEMory?

Description	Returns the data in acquisition memory for the selected channel as a header + raw data.				
Syntax	:ACquire<X>:MEMory?				
Related Commands	:ACquire:RECOrdlength :HEADer				
Parameter	<X> Channel number (1 to 4)				
Return parameter	<table style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 15%; border-right: 1px solid black;"><string></td> <td>Returns acquisition settings followed by raw waveform block data.</td> </tr> <tr> <td style="border-right: 1px solid black;"><waveform block data></td> <td> <string> Returns the acquisition settings for the selected channel. Format: parameter(1),setting(1);parameter(2),setting(2)...parameter(n),setting(n);Waveform Data; <waveform block data> Header followed by the raw waveform data. </td> </tr> </table>	<string>	Returns acquisition settings followed by raw waveform block data.	<waveform block data>	<string> Returns the acquisition settings for the selected channel. Format: parameter(1),setting(1);parameter(2),setting(2)...parameter(n),setting(n);Waveform Data; <waveform block data> Header followed by the raw waveform data.
<string>	Returns acquisition settings followed by raw waveform block data.				
<waveform block data>	<string> Returns the acquisition settings for the selected channel. Format: parameter(1),setting(1);parameter(2),setting(2)...parameter(n),setting(n);Waveform Data; <waveform block data> Header followed by the raw waveform data.				

Format:

Header: The header (in ASCII) encodes the number of bytes for the header followed by the number of data points in bytes for the raw data.



Raw Data:

Each two bytes (in hex) encodes the vertical data of a data point. The data is signed hex data (2's complement, -32768 ~ 32767).

Waveform Raw Data Example:

Header raw data.....

Hex:

23 34 32 30 30 30 00 1C 00 1B 00 1A 00
1A 00 1B

ASCII/Decimal:

#42000 28 27 26 26 27.....

The actual value of a data point can be calculated with the following formula:
(Decimal value of hex data / AD Factor) * vertical scale.

Note: AD Factor is fixed as 25. The vertical scale is returned with the

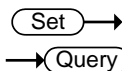
acquisition settings that precede the raw data.

For example if the raw data for a point is 001C (=28 decimal) then,
 $(28/25) \times 0.5 = 0.56V$

Example :ACQuire1:MEMory?
 Format,2.0E;Memory Length,10000;IntpDistance,0;
 Trigger Address,4999;Trigger Level,1.160E+01;
 Source,CH1;Vertical Units,V;Vertical Units
 Div,0;Vertical Units Extend
 Div,15;Label,ACK ;Probe Type,0;Probe
 Ratio,1.000e+01;Vertical Scale,5.000e+00;Vertical
 Position,-1.100e+01;Horizontal Units,S;Horizontal
 Scale,5.000E-04;Horizontal Position,0.000E+00;
 Horizontal Mode,Main;SincET Mode,Real
 Time;Sampling Period,5.000e-07;Horizontal Old
 Scale,5.000E-04;Horizontal Old Position,0.000E+00;
 Firmware,V0.99b8;Time,02-Oct-14 17:00:43;
 Waveform Data;
 #520000.....follows waveform block
 data in hex.

Note You need to set the command “:USBDelay ON” once to avoid missing data before running the program if you use USB interface to transfer data in the Windows 10 operating system.

:ACQuire:FILTer:SOURce



Description	Returns the source of the filter.
Syntax	:ACQuire:FILTer:SOURce {CH1 CH2 CH3 CH4 ?}
Parameter/ Return parameter	CH1 ~ CH4 Source channel

Example :ACQuire:FILTer:SOURce?
 CH1
 Sets the filter source to CH1.

Set →
 → Query

:ACQUIRE:FILTER

Description	Turns the filter on/off or queries its status.	
Syntax	:ACQUIRE:FILTER {ON OFF ?}	
Parameter/ Return parameter	ON	Filter on.
	OFF	Filter off.
Example	:ACQUIRE:FILTER? OFF Indicates that the filter is turned off.	

Set →
 → Query

:ACQUIRE:FILTER:FREQUENCY

Description	Sets or queries the filter frequency.	
Syntax	:ACQUIRE:FILTER:FREQUENCY {DEFAULT <NRf> ?}	
Parameter/ Return parameter	DEFAULT	Sets the filter frequency to the default.
	<NRf>	Manually sets the filter frequency. (1Hz ~ 500MHz)
Example	:ACQUIRE:FILTER:FREQUENCY 1 Sets the filter frequency to 1Hz.	

Set →
 → Query

:ACQUIRE:FILTER:TRACKING

Description	Turns filter tracking on/off or queries its state.	
Syntax	:ACQUIRE:FILTER:TRACKING {ON OFF ?}	
Parameter/ Return parameter	OFF	Tracking off
	ON	Tracking on
Example	:ACQUIRE:FILTER:TRACKING ON Turns filter tracking on.	

:ACquire<X>:STATe? → Query

Description	Returns the status of waveform data.	
Syntax	:ACquire<X>:STATe?	
Parameter	<X>	Channel number (1 to 4)
Return parameter	0	Raw data is not ready
	1	Raw data is ready
Example	:ACquire1:STATe? 0 Returns 0. Channel 1's raw data is not ready. <i>Note: If the oscilloscope changes the acquisition status from STOP to RUN, the status will be reset as zero.</i>	

Set →

:ACquire:INTERpolation → Query

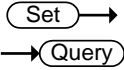
Description	Selects or returns the interpolation mode.	
Syntax	:ACquire:INTERpolation {ET SINC ?}	
Parameter/Return parameter	ET	Equivalent Time interpolation. The MDO-2000EG/2000EX doesn't support ET.
	SINC	Sets to SIN(X)/X interpolation
Example	:ACquire:INTERpolation? >SINC Returns SINC as the interpolation mode.	

Set →

:ACquire:RECOrdlength → Query

Description	Sets or queries the record length.	
Syntax	:ACquire:RECOrdlength {<NRf> ?}	
Parameter/Return parameter	<NRf>	Record length. Settable record length: (1e+3 1e+4 1e+5 1e+6 1e+7)

Example :ACQuire:RECOrdlength 1e+3
 Sets the record length to 1000 points.

:HEADer


Description Configures whether the returned data of the :ACQuire:MEM query will contain header information or not. It is set to ON by default.

Syntax :HEADer {OFF | ON | ?}

Related Commands :ACQuire<X>:MEMory?

Parameter	ON	Add header information.
	OFF	Don't add header information.

Return parameter Returns the configuration (ON, OFF) for the selected channel.

Example :HEADer ON

Autoscale Commands

:AUTOSet	45
:AUTORSET:MODE	45

:AUTOSet

Set →

Description Runs the Autoset function to automatically configure the horizontal scale, vertical scale, and trigger according to the input signal.

Syntax :AUTOSet

Set →

:AUTORSET:MODE

→ Query

Description Sets the Autoset mode or queries its state.

Syntax :AUTORSET:MODE {FITScreen | ACPriority | ?}

Related Commands :AUTOSet

Parameter/Return parameter	FITScreen	Fit Screen mode
	ACPriority	AC priority mode

Example :AUTORSET?
FITSCREEN

Vertical Commands

:CHANnel<X>:BWLimit.....	46
:CHANnel<X>:COUPling	47
:CHANnel<X>:DESKew	47
:CHANnel<X>:DISPlay	47
:CHANnel<X>:EXPand	48
:CHANnel<X>:IMPedance?.....	48
:CHANnel<X>:INVert	49
:CHANnel<X>:POSition	49
:CHANnel<X>:PROBe:RATio	50
:CHANnel<X>:PROBe:TYPe.....	50
:CHANnel<X>:SCALE	50

:CHANnel<X>:BWLimit




Description	Sets or returns the bandwidth limit on/off.	
Syntax	:CHANnel<X>:BWLimit {FULL <NR3> ?}	
Parameter	<X>	Channel 1,2,3,4
	FULL	Full bandwidth
	<NR3>	Sets the bandwidth limit to a pre-defined bandwidth.
		100E+6: 100MHz
		20E+6: 20MHz
Return Parameter	<NR3>	Returns the bandwidth.
	Full	Full bandwidth
Example	:CHANnel1:BWLimit 2.000E+07 Sets the channel 1 bandwidth to 20MHz.	

:CHANnel<X>:COUPling
 →
 →

Description	Selects or returns the coupling mode.	
Syntax	CHANnel<X>:COUPling {AC DC GND ?}	
Parameter	<X>	Channel 1,2,3,4
	AC	AC coupling
	DC	DC coupling
	GND	Ground coupling
Return parameter	Returns the coupling mode.	
Example	:CHANnel1:COUPling DC Sets the coupling to DC for Channel 1.	

:CHANnel<X>:DESKew
 →
 →

Description	Sets the deskew time in seconds.	
Syntax	:CHANnel<X>:DESKew { <NR3> ?}	
Parameter	<X>	Channel 1,2,3,4
	<NR3>	Deskew time: -5.00E -11 to 5.00E-11 -50ns to 50 ns. (10 ps /step)
Return parameter	<NR3>	Returns the deskew time.
Example	:CHANnel1:DESKew 1.300E-9 Sets the deskew time to 1.3 nano seconds.	

:CHANnel<X>:DISPlay
 →
 →

Description	Turns a channel on/off or returns its status.	
Syntax	:CHANnel<X>:DISPlay {OFF ON ?}	
Parameter	<X>	Channel 1,2,3,4
	OFF	Channel off
	ON	Channel on

Return Parameter	ON	Channel is on
	OFF	Channel is off

Example :CHANnel1:DISPlay ON
Turns on Channel 1

Set →

→ Query

:CHANnel<X>:EXPand

Description Sets Expand By Ground or Expand By Center for a channel or queries its status.

Syntax :CHANnel<X>:EXPand {GND | CENTER | ?}

Parameter	<X>	Channel 1,2,3,4
	GND	Ground
	CENTER	Center

Return parameter	GND	Expand By Ground
	CENTER	Expand By Center

Example :CHANnel1:EXPand GND
Sets Channel 1 to Expand By Ground.

:CHANnel<X>:IMPedance?

→ Query

Description Returns the impedance of the oscilloscope. (The impedance of the MDO-2000EG/2000EX is fixed at 1MΩ)

Syntax :CHANnel<X>:IMPedance?

Parameter	<x>	Channel
	1/2/3/4	CH1/2/3/4

Return parameter <NR3> Returns the impedance value.

Example :CHANnel1:IMPedance?
1.000000E+06
The impedance is 1M ohms.

:CHANnel<X>:INVert




Description	Inverts a channel or returns its status.	
Syntax	:CHANnel<X>:INVert {OFF ON ?}	
Parameter	<X>	Channel 1, 2, 3, 4
	OFF	Invert off
	ON	Invert on
Return parameter	ON	Invert on
	OFF	Invert off
Example	:CHANnel1:INVert ON Inverts Channel 1	

:CHANnel<X>:POSition




Description	Sets or returns the position level for a channel.	
Note	<p>The vertical position will only be set to closest allowed value. The position level range depends on the vertical scale.</p> <p>The scale must first be set before the position can be set.</p>	
Syntax	:CHANnel<X>:POSition { <NRf> ?}	
Parameter	<X>	Channel 1, 2, 3, 4
	<NRf>	Position. Range depends on the vertical scale.
Return parameter	<NR3>	Returns the position value.
Example 1	:CHANnel1:POSition 2.4E-3 Sets the Channel 1 position to 2.4mV/mA	
Example 2	:CHANnel1:POSition? 2.4E-3 Returns 2.4mV as the vertical position.	

Set →
 → Query

:CHANnel<X>:PROBe:RATio

Description	Sets or returns the probe attenuation factor.	
Syntax	:CHANnel<X>:PROBe:RATio { <NRf> ?}	
Related Commands	:CHANnel<X>:PROBe:TYPE	
Parameter	<X>	Channel 1, 2, 3, 4
	<NRf>	Probe attenuation factor
Return parameter	<NR3>	Returns the probe factor
Example	:CHANnel1:PROBe:RATio 1.00E+0 Sets the Channel 1 probe attenuation factor to 1x	

Set →
 → Query

:CHANnel<X>:PROBe:TYPE

Description	Sets or returns the probe type (voltage/current).	
Syntax	:CHANnel<X>:PROBe:TYPE { VOLTage CURRent ?}	
Related Commands	:CHANnel<X>:PROBe:RATio	
Parameter	<X>	Channel 1, 2, 3, 4
	VOLTage	Voltage
	CURRent	Current
Return parameter	Returns the probe type.	
Example	:CHANnel1:PROBe:TYPE VOLTage Sets the Channel 1 probe type to voltage.	

Set →
 → Query

:CHANnel<X>:SCALE

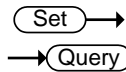
Description	Sets or returns the vertical scale. The scale depends on the probe attenuation factor. Note the probe attenuation factor should be set before the scale.	
-------------	---	--

Syntax	:CHANnel<X>:SCALE { <NRf> ?}	
Parameter	<X>	Channel 1, 2, 3, 4
	<NRf>	Vertical scale: 2e-3 to 1e+1 2mV to 10V (Probe x1)
Return parameter	<NR3>	Returns the vertical scale in volts or amps.
Example	:CHANnel1:SCALE 2.00E-2	
	Sets the Channel 1 vertical scale to 20mV/div	


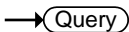
Math Commands



:MATH:DISP	52
:MATH:TYPe	53
:MATH:DUAL:SOURce<X>	53
:MATH:DUAL:OPERator	53
:MATH:DUAL:POSition	54
:MATH:DUAL:SCALe	54
:MATH:FFT:SOURce	55
:MATH:FFT:MAG	55
:MATH:FFT:WINDow	55
:MATH:FFT:POSition	56
:MATH:FFT:SCALe	56
:MATH:FFT:HORizontal:SCALe	57
:MATH:FFT:HORizontal:POSition	57
:MATH:DEFine	57
:MATHVAR?	58
:MATHVAR:VAR<X>	59
:MATH:ADVanced:POSition	59
:MATH:ADVanced:SCALe	59

:MATH:DISP



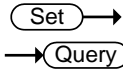
Description	Turns the math display on or off on the screen.	
Syntax	:MATH:DISP {OFF ON ?}	
Parameter/ Return parameter	OFF	Math is not displayed on screen
	ON	Math is displayed on screen
Example	:MATH:DISP OFF Math is off.	

		 
:MATH:TYPE		
Description	Queries or sets the Math type to FFT, Advanced Math or to dual channel math operations	
Syntax	:MATH:TYPE { DUAL ADVanced FFT ? }	
Related Commands	:MATH:DISP	
Parameter	DUAL	Dual channel operations
	ADVanced	Advanced math operations
	FFT	FFT operations
Return parameter	Returns the math type.	
Example	:MATH:TYPE DUAL Sets the Math type to dual channel math operation.	

		 
:MATH:DUAL:SOURce<X>		
Description	Sets the dual math source for source 1 or 2.	
Syntax	:MATH:DUAL:SOURce<X> { CH1 CH2 CH3 CH4 REF1 REF2 REF3 REF4 ? }	
Parameter	<X>	Source number 1 or 2
	CH1~4	Channel 1 to 4
	REF1~4	Reference waveforms 1 to 4
Return parameter	Returns the source for the source 1 or 2.	
Example	:MATH:DUAL:SOURce1 CH1 Sets source1 as channel 1.	

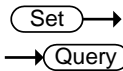
		 
:MATH:DUAL:OPERator		
Description	Sets the math operator for dual math operations.	

Syntax	:MATH:DUAL:OPERator {PLUS MINUS MUL DIV ?}	
Parameter	PLUS	+ operator
	MINUS	- operator
	MUL	× operator
	DIV	÷ operator
Return parameter	Returns operator type.	
Example	:MATH:DUAL:OPERator PLUS Sets the math operator as plus (+).	



:MATH:DUAL:POSition

Description	Sets the vertical position of the displayed math result expressed by unit/division.	
Syntax	:MATH:DUAL:POSition {<NRf> ? }	
Parameter	<NRf>	Vertical position Depends on the vertical scale (Unit/Div)
Return parameter	<NR3>	Returns the vertical position.
Example	:MATH:DUAL:POSition 1.0E+0 Sets the vertical position to 1.00 unit/div.	



:MATH:DUAL:SCALe

Description	Sets the vertical scale of the displayed math result.	
Syntax	:MATH:DUAL:SCALe {<NRf> ? }	
Parameter	<NRf>	Vertical scale
Return parameter	<NR3>	Returns the scale.
Example	:MATH:DUAL:SCALe 2.0E-3 Sets the vertical scale to 2mV/2mA.	

:MATH:FFT:SOURce




Description	Sets and queries the FFT math source.	
Syntax	:MATH:FFT:SOURce { CH1 CH2 CH3 CH4 REF1 REF2 REF3 REF4 ? }	
Related commands	:MATH:ADVanced:EDIT:SOURce<X> :MATH:ADVanced:EDIT:OPERator	
Parameter	CH1~4 REF1~4	Channel 1 to 4 Reference waveform 1 to 4
Return parameter	Returns the FFT source.	
Example	:MATH:FFT:SOURce CH1 Sets the FFT math source as channel 1.	

:MATH:FFT:MAG




Description	Sets FFT vertical units as linear or decibels.	
Syntax	:MATH:FFT:MAG {LINEAR DB ?}	
Parameter	LINEAR DB	Linear units (Vrms) Logarithmic units (dB)
Return parameter	Returns the FFT vertical units.	
Example	:MATH:FFT:MAG DB Sets FFT vertical units to dB.	

:MATH:FFT:WINDow




Description	Sets the windowing filter used for the FFT function.	
Syntax	:MATH:FFT:WINDow {RECTangular HAMming HANning BLAckman ?}	
Parameter	RECTangular	Rectangular window

HAMming	Hamming window
HANning	Hanning window
BLAckman	Blackman window

Return parameter Returns the FFT window.

Example :MATH:FFT:WINDow HAMming
Sets the FFT window filter to hamming.

Set →

→ Query

:MATH:FFT:POSition

Description Sets the vertical position of the displayed FFT result.

Syntax MATH:FFT:POSition { <NRf> | ? }

Parameter <NRf> Vertical position: -12e+0 to +12e+0 (12 units/division to +12 units/division.)

Return parameter <NR3> Returns the vertical position.

Example :MATH:FFT:POSition -2e-1
Sets the FFT position to -0.2 divisions.

Set →

→ Query

:MATH:FFT:SCALE

Description Sets the vertical scale of the displayed FFT result.

Syntax :MATH:FFT:SCALE {<NRf> | ?}

Parameter <NRf> Vertical scale:
Linear: 2e-3 to 1e+3 (2mV~1kV)
dB: 1e+0 to 2e+1 (1~20dB)

Return parameter <NR3> Returns vertical scale.

Example :MATH:FFT:SCALE 1.0e+0
Sets the scale to 1dB.

Set →
 → Query

:MATH:FFT:HORizontal:SCALe

Description Sets or queries the zoom scale for FFT math.

Syntax :MATH:FFT:HORizonatal:SCALe {<NRf> | ?}

Parameter <NRf> Zoom scale: 1 to 20 times

Return parameter <NR3> Returns zoom scale.

Example :MATH:FFT:HORizontal:SCALe 5
 Sets the zoom scale to 5X.

Set →
 → Query

:MATH:FFT:HORizontal:POSition

Description Sets the horizontal position of the displayed FFT result.

Syntax MATH:FFT:HORizontal:POSition { <NRf> | ? }

Parameter <NRf> Horizontal position: 0Hz ~ 999.9kHz

Return parameter <NR3> Returns the vertical position.

Example :MATH:FFT:HORizontal:POSition 6.0e5
 Sets the FFT horizontal position to 600kHz.

Set →
 → Query

:MATH:DEFine

Description Sets or queries the advanced math expression as a string.

Syntax :MATH:DEFine {<string> | ?}

Related :MATH:DISP
 :MATH:TYPE

Parameter <string> An expression enclosed in double quotes. Note, ensure parentheses are used correctly in the expression. The expression can contain the following parts:

Source	CH1~CH4, Ref1~Ref4
Function	Intg(, Diff(, log(, ln(, Exp(, Sqrt(, Abs(, Rad(, Deg(, sin(, cos(, tan(, asin(, acos(, atan(
Variable	VAR1, VAR2
Operator	+, -, *, /, (,), !(, <, >, <=, >=, ==, !=, , &&
Figure	0, 1, 2, 3, 4, 5, 6, 7, 8, 9, ., E
Measurement	Pk-Pk(, Max(, Min(, Amp(, High(, Low(, Mean(, CycleMean(, RMS(, CycleRMS(, Area(, CycleArea(, ROVShoot(, FOVShoot(, Freq(, Period(, Rise(, Fall(, PosWidth(, NegWidth(, Dutycycle(, FRR(, FRF(, FFR(, FFF(, LRR(, LRF(, LFR(, LFF(, Phase(

Return parameter Returns the expression as a string.

Example :MATH:DISP ON
 :MATH:TYPe ADVanced
 MATH:DEFine "CH1-CH2"
 Sets the math expression to CH1-CH2.

:MATHVAR?



Description	Returns the value of the VAR1 and VAR2 variables.
Syntax	MATHVAR?
Related Commands	MATHVAR:VAR<X> MATH:DEFine
Return parameter	<string> VAR1 <NR3>; VAR2 <NR3>

Example MATHVAR?
 VAR1 1.000000E+06; VAR2 1.0E+1
 Returns the value of both variables.

Set →
 → Query

Description Sets or returns the VAR1 or VAR2 variables.

Syntax MATHVAR:VAR<x> {<NRf> | ?}

Related MATH:DEFine
 Commands

Parameter	<x>	1, 2 (VAR1 or VAR2)
	<NRf>	Value of VAR1/VAR2

Return parameter	<NR3>	Returns the value of VAR1/VAR2
------------------	--------------------	--------------------------------

Example :MATH:VAR1 6.0e4
 Sets VAR1 to 60000.

Set →
 → Query

Description Sets the vertical position of the advanced math result, expressed in unit/div.

Syntax MATH:ADVanced:POSition { <NRf> | ? }

Parameter	<NRf>	Vertical position: -12e+0 to +12e+0 (12 units/division to +12 units/division.)
-----------	--------------------	---

Return parameter	<NR3>	Returns the vertical position.
------------------	--------------------	--------------------------------

Example :MATH:ADVanced:POSition 1.0e+0
 Sets the position as 1.00 unit/div.

Set →
 → Query

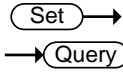
Description Sets or queries the vertical scale the advanced math result.

Syntax :MATH:ADVanced:SCALE {<NRf> | ?}

Parameter	<NRf>	Vertical scale
Return parameter	<NR3>	Returns the vertical scale.
Example	:MATH:ADVanced:SCALE 2.0E-3 Sets the vertical scale to 2mV/Div.	

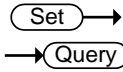
Cursor Commands

:CURSor:MODE	62
:CURSor:SOURce.....	62
:CURSor:HUNI	63
:CURSor:HUSE	63
:CURSor:VUNI	64
:CURSor:VUSE	64
:CURSor:DDT	64
:CURSor:H1Position	65
:CURSor:H2Position	65
:CURSor:HDELta	66
:CURSor:V1Position.....	66
:CURSor:V2Position.....	66
:CURSor:VDELta	67
:CURSor:XY:RECTangular:X:POSition<X>	67
:CURSor:XY:RECTangular:X:DELta	67
:CURSor:XY:RECTangular:Y:POSition<X>	68
:CURSor:XY:RECTangular:Y:DELta	68
:CURSor:XY:POLar:RADIUS:POSition<X>.....	68
:CURSor:XY:POLar:RADIUS:DELta	69
:CURSor:XY:POLar:THETA:POSition<X>	69
:CURSor:XY:POLar:THETA:DELta	69
:CURSor:XY:PRODuct:POSition<X>	70
:CURSor:XY:PRODuct:DELta.....	70
:CURSor:XY:RATio:POSition<X>.....	70
:CURSor:XY:RATio:DELta	71




:CURSor:MODE

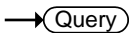
Description	Sets cursor mode to horizontal (H) or horizontal and vertical (HV). Note: When the cursor source is set to bus, then only the horizontal cursor is available.	
Syntax	:CURSor:MODE {OFF H HV ? }	
Parameter	OFF	Turns the cursors off.
	H	Turns the horizontal cursors on.
	HV	Turns horizontal and vertical cursors on.
Return parameter	Returns the state of the cursors (H, HV, OFF).	
Example	:CURSor:MODE OFF Turns the cursors off.	



:CURSor:SOURce


Description	Sets or queries the cursor source.	
Syntax	:CURSor:SOURce {CH1 CH2 CH3 CH4 REF1 REF2 REF3 REF4 MATH LOGic BUS1 ? }	
Parameter	CH1~CH4	Channel 1 to 4
	REF1~4	Reference waveform 1 to 4
	MATH	Math source
	LOGic	Logic source
	BUS1	Bus source
Return parameter	Returns the cursor source.	
Example	:CURSor:SOURce CH1 Turns the cursor source as channel 1.	





:CURSor:HUNI

Description	Sets or queries the units for the horizontal bar cursors.	
Syntax	:CURSor:HUNI {SEConds HERTz DEGrees PERcent ?}	
Related Commands	:CURSor:MODe	
Parameter	SEConds	Sets the cursor units to time in seconds.
	HERtz	Sets the cursor units to frequency.
	DEGrees	Sets the cursor units to degrees.
	PERcent	Sets the cursor units to percent.
Return parameter	Returns the unit type.	
Example	:CURSor:HUNI SEConds Sets the units to time in seconds.	



:CURSor:HUSE

Description	Sets the current cursor position as the phase or ratio reference for the Percent or Degrees (horizontal) cursors.	
Note	This command can only be used when :CURSor:HUNI is set to DEGrees or PERcent.	
Syntax	:CURSor:HUSE {CURRent}	
Related Commands	:CURSor:MODe :CURSor:HUNI	
Parameter	CURRent	Uses the current horizontal position
Example	:CURSor:HUSE CURRent.	

Set →

→ Query

:CURSor:VUNI

Description	Sets or queries the units for the vertical bar cursors.	
Syntax	:CURSor:VUNI {BASE PERcent ?}	
Related Commands	:CURSor:MODE	
Parameter	BASE	Sets the vertical cursor units the same as the scope units (V or A).
	PERcent	Sets the displayed units to percent.
Return parameter	Returns the unit type.	
Example	:CURSor:VUNI BASE Sets the units to the base units.	

Set →

:CURSor:VUSE

Description	Sets the current cursor position as the ratio reference for the Percent (vertical) cursors.	
Note	This command can only be used when :CURSor:VUNI is set to PERcent.	
Syntax	:CURSor:VUSE {CURRent}	
Related Commands	:CURSor:MODE :CURSor:VUNI	
Parameter	CURRent	Uses the current vertical position
Example	:CURSor:VUSE CURRent.	

→ Query

:CURSor:DDT

Description	Returns the deltaY/deltaT (dy/dT) readout. This function is only supported if the source channels are CH1~4, Ref1~4 or Math.	
Syntax	:CURSor:DDT{?}	

Related Commands :CURSor:MODE

Return Parameter <NR3> Returns the readout in <NR3> format.

Example :CURSor:DDT?
4.00E-05

:CURSor:H1Position

Set →

→ Query

Description Sets or returns the first horizontal cursor (H1) position.

Syntax :CURSor:H1Position {<NRf> | ?}

Related Commands :CURSor:H2Position

Parameter <NRf> Horizontal position

Return parameter Returns the cursor position.

Example :CURSor:H1Position?
-1.34E-3
Returns the H1 cursor position as -1.34ms.

Set →

→ Query

:CURSor:H2Position

Description Sets or returns the second horizontal cursor (H2) position.

Syntax :CURSor:H2Position {<NRf> | ?}

Related Commands :CURSor:H1Position

Parameter <NRf> Horizontal Position

Return parameter Returns the cursor position.

Example :CURSor:H2Position 1.5E-3
Sets the H2 cursor position to 1.5ms.

:CURSor:HDELta

→ Query

Description	Returns the delta of H1 and H2.	
Syntax	:CURSor:HDELta{?}	
Return Parameter	<NR3>	Returns the distance between two horizontal cursors.
Example	:CURSor:HDELta? 5.0E-9 Returns the horizontal delta as 5ns.	

Set →

:CURSor:V1Position

→ Query

Description	Sets the first vertical cursor (V1) position.	
Syntax	:CURSor:V1Position {<NRf> ?}	
Parameter	<NRf>	Vertical position. Depends on the vertical scale.
Return parameter	<NR3>	Returns the cursor position.
Example	:CURSor:V1Position 1.6E -1 Sets the V1 cursor position to 160mA.	

Set →

:CURSor:V2Position

→ Query

Description	Sets the first vertical cursor (V2) position.	
Syntax	:CURSor:V2Position {<NRf> ?}	
Parameter	<NRf>	Vertical position. Depends on the vertical scale.
Return parameter	<NR3>	Returns the cursor position.
Example	:CURSor:V2Position 1.1E-1 Sets the V2 cursor position to 110mA.	

:CURSor:VDELta → Query

Description	Returns the delta of V1 and V2.	
Syntax	:CURSor:VDELta{?}	
Return Parameter	<NR3>	Returns the difference between two vertical cursors.
Example	:CURSor:VDELta? 4.00E+0 Returns the vertical delta as 4 volts.	

:CURSor:XY:RECTangular:X:POSition<X> Set →
→ Query

Description	Sets or queries the horizontal position in XY mode for the X rectangular coordinates for cursor 1 or 2.	
Syntax	:CURSor:XY:RECTangular:X:POSition<X> {<NRf> ?}	
Parameter	<X>	Cursor 1, 2
	<NRf>	Horizontal position co-ordinates
Return parameter	<NR3>	Returns the cursor position.
Example	:CURSor:XY:RECTangular:X:POSition1 4.0E-3 Sets the X-coordinate cursor 1 position to 40mV/mV.	

:CURSor:XY:RECTangular:X:DELta → Query

Description	Returns the delta value of cursor 1 and 2 on the X coordinate.	
Syntax	:CURSor:XY:RECTangular:X:DELta{?}	
Return Parameter	<NR3>	Returns the delta value of cursor 1 and 2 as <NR3>.

Example :CURSor:XY:RECTangular:X:DELta?
 80.0E-3
 Returns the horizontal delta as 80mV.

Set →

:CURSor:XY:RECTangular:Y:POSition<X> → Query

Description Sets or queries the vertical position in XY mode for the Y rectangular coordinates for cursor 1 or 2.

Syntax :CURSor:XY:RECTangular:Y:POSition<X> {<NRf>|?}

Parameter	<X>	Cursor 1, 2
	<NRf>	Vertical position co-ordinates

Return parameter	<NR3>	Returns the cursor position.
------------------	-------	------------------------------

Example :CURSor:XY:RECTangular:Y:POSition1 4.0E-3
 Sets the Y-coordinate cursor 1 position to 40mV/mV.

:CURSor:XY:RECTangular:Y:DELta → Query

Description Returns the delta value of cursor 1 and 2 on the Y coordinate.

Syntax :CURSor:XY:RECTangular:Y:DELta{?}

Return Parameter	<NR3>	Returns the delta value of cursor 1 and 2 as <NR3>.
------------------	-------	---

Example :CURSor:XY:RECTangular:Y:DELta?
 80.0E-3
 Returns the horizontal delta as 80mV.

:CURSor:XY:POLar:RADIUS:POSition<X> → Query

Description Queries the polar radius position for the specified cursor in XY mode, where X can be either cursor 1 or 2.

Syntax	:CURSor:XY:POLar:RADIUS:POSition<X>{?}
Parameter	<X> 1, 2 (cursor 1, cursor 2)
Return Parameter	<NR3> Returns the polar radius position.
Example	:CURSor:XY:POLar:RADIUS:POSition1? 80.0E-3 Returns the polar radius position as 80.0mV.

:CURSor:XY:POLar:RADIUS:DELta → Query

Description	Returns the radius delta value of cursor 1 and 2.
Syntax	:CURSor:XY:POLar:RADIUS:DELta{?}
Return Parameter	<NR3> Returns the radius delta.
Example	:CURSor:XY:POLar:RADIUS:DELta? 31.4E-3 Returns the radius delta as 31.4mV.

:CURSor:XY:POLar:THETA:POSITION<X> → Query

Description	Queries the polar angle for the specified cursor in XY mode, where X can be either 1 or 2.
Syntax	:CURSor:XY:POLar:THETA:POSITION<X>{?}
Parameter	<X> 1, 2 (Cursor 1, Cursor 2)
Return parameter	<NR3> Returns the polar angle.
Example	:CURSor:XY:POLar:RADIUS:POSition1? 8.91E+1 Returns the polar angle for cursor1 as 89.1°.

:CURSor:XY:POLar:THETA:DELta → Query

Description	Queries the polar angle delta between cursor1 and cursor2.
-------------	--

Syntax	:CURSor:XY:POLar:THETA:DELta{?}	
Return parameter	<NR3>	Returns the theta delta between cursor1 and cursor2.
Example	:CURSor:XY:POLar:THETA:DELta? 9.10E+0 Returns the delta as 9.1°.	

:CURSor:XY:PRODUct:POSItion<X> → Query

Description	Queries the product in XY mode for the specified cursor, where x can be either 1 or 2.	
Syntax	:CURSor:XY:PRODUct:POSItion<X>{?}	
Parameter	<X>	1, 2 (Cursor 1, Cursor 2)
Return parameter	<NR3>	Returns the product value of the Cursor1 or Cursor2.
Example	:CURSor:XY:PRODUct:POSItion1? 9.44E-5 Returns the product of cursor1 as 94.4uVV.	

:CURSor:XY:PRODUct:DELta → Query

Description	Queries the product delta in XY mode.	
Syntax	:CURSor:XY:PRODUct:DELta{?}	
Return parameter	<NR3>	Returns the product delta.
Example	:CURSor:XY:PRODUct:DELta? 1.22E-5 Returns the product delta as 12.2uVV.	

:CURSor:XY:RATIo:POSItion<X> → Query

Description	Queries the ratio in XY mode for the specified cursor, where x can be either cursor 1 or 2.	
-------------	---	--

Syntax	:CURSor:XY:RATio:POSition<X>{?}	
Parameter	<X>	1, 2 (Cursor 1, Cursor 2)
Return parameter	<NR3>	Returns the ratio.
Example	:CURSor:XY:RATio:POSition? 6.717E+1 Returns the ratio value as 6.717V/V.	

:CURSor:XY:RATio:DELta → Query

Description	Queries the ratio delta in XY mode.	
Syntax	:CURSor:XY:RATio:DELta{?}	
Return parameter	<NR3>	Returns the ratio delta.
Example	:CURSor:XY:RATio:DELta? 5.39E+1 Returns the ratio delta as 53.9V/V.	

Display Commands

:DISPlay:INTensity:WAVEform..... 72
 :DISPlay:INTensity:GRATicule 72
 :DISPlay:INTensity:BACKLight 73
 :DISPlay:INTensity:BACKLight:AUTODim:Enable 73
 :DISPlay:INTENSITY:BACKLight:AUTODim:TIME 73
 :DISPlay:PERSistence 74
 :DISPlay:GRATicule 74
 :DISPlay:WAVEform 75
 :DISPlay:OUTPut 75

:DISPlay:INTensity:WAVEform (Set) →
← (Query)

Description	Sets or queries the waveform intensity level.	
Syntax	:DISPlay:INTensity:WAVEform {<NRf> ?}	
Parameter	<NRf>	0.0E+0 to 1.0E+2 (0% to 100%)
Return Parameter	<NR3>	Returns the intensity.
Example	:DISPlay:INTensity:WAVEform 5.0E+1 Sets the waveform intensity to 50%.	

:DISPlay:INTensity:GRATicule (Set) →
← (Query)

Description	Sets or queries the graticule intensity level.	
Syntax	:DISPlay:INTensity:GRATicule {<NRf> ?}	
Parameter	<NRf>	1.0E+0 to 1.0E+2 (10% to 100%)
Return Parameter	<NR3>	Returns the graticule intensity.
Example	:DISPlay:INTensity:GRATicule 5.0E+1 Sets the graticule intensity to 50%.	

:DISPlay:INTensity:BACKLight  


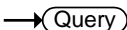
Description Sets or queries the intensity of the backlight display.

Syntax :DISPlay:INTensity:BACKLight {<NRf> | ?}

Parameter <NRf> 1.0E+0 to 1.0E+2 (10% to 100%)

Return Parameter <NR3> Returns the backlight intensity.

Example :DISPlay:INTensity:BACKLight 5.0E+1
Sets the backlight intensity to 50%.

:DISPlay:INTensity:BACKLight:AUTODim  
:ENABLE

Description Sets or queries the display auto-dim function.

Syntax :DISPlay:INTensity:BACKLight:AUTODim:ENABLE {OFF | ON | ?}

**Parameter/
Return parameter** OFF Turn auto-dim on.
ON Turn auto-dim off.

Example :DISPlay:INTensity:BACKLight:AUTODim:ENABLE ON
Turns the auto-dim function on.

:DISPlay:INTENSITY:BACKLight:AUTODim  
:TIME

Description Sets or queries the display auto-dim time.

Syntax :DISPlay:INTensity:BACKLight:AUTODim:TIME {<NR1> | ?}

**Parameter/
Return parameter** <NR1> 1 ~ 180 minutes. Time in minutes.

Example :DISPlay:INTensity:BACKLight:AUTODim:TIME 10
Sets the auto-dim time to 10 minutes.

Set →

→ Query






:DISPlay:PERsistence

Description	Sets or queries the waveform persistence level.	
Syntax	:DISPlay:PERsistence { INFINite OFF <NRf> ? }	
Parameter	<NRf>	1.6E-2 ~ 4.0E+0. (16mS to 10S) Range(1.6E-2, 30E-3, 60E-3, 120E-2, 240E-3, 500E-3, 750E-3, 1, 1.5,2,...,9.5,10
	INFINite	Infinite persistence
	OFF	No persistence
Return Parameter	<NR3>	Returns the persistence time.
	INFINite	Infinite persistence
	OFF	No persistence
Example	:DISPlay:PERsistence 2.0E+0 Sets the persistence to 2 seconds.	

Set →

→ Query

:DISPlay:GRATicule

Description	Sets or queries graticule display type.	
Syntax	:DISPlay:GRATicule {FULL GRID CROSSs FRAME ?}	
Parameter	FULL	
	FRAME	
Parameter	CROSSs	
	GRID	
Return parameter	Returns the graticule type.	
Example	:DISPlay:GRATicule FULL Sets the graticule to  .	

:DISPlay:WAVEform



Description	Sets or queries whether the waveforms are drawn as vectors or dots.	
Syntax	:DISPlay:WAVEform {VECTor DOT ?}	
Parameter	VECTor	Vectors
	DOT	Dots
Return parameter	Returns VECTOR or DOT.	
Example	:DISPlay:WAVEform VECTor Sets the waveform to vectors.	

:DISPlay:OUTPut 

Description	Returns the screen image as a 16 bit RGB run length encoded image.
Syntax	:DISPlay:OUTPut{?}
Return parameter	Format: header+data+LF For example assuming the image data size is 60072 bytes then the following would be returned: #560072<[count] [color] [count] [color]..... ><LF> Where #560072 is the header, each [count] and [color] data are 2 bytes and <LF> is a line feed character.
Note	You need to set the command “:USBDelay ON” once to avoid missing data before running the program if you use USB interface to transfer data in the Windows 10 operating system.

Hardcopy Commands

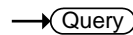
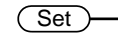
:HARDcopy:START	76
:HARDcopy:MODE	76
:HARDcopy:PRINTINKSaver	77
:HARDcopy:SAVEINKSaver	77
:HARDcopy:SAVEFORMat	77
:HARDcopy:ASSIGN	78

:HARDcopy:START



Description	Executing the HARDcopy:START command is the equivalent of pressing the Hardcopy key on the front panel.
Syntax	:HARDcopy:START
Related Commands	:HARDcopy:MODE :HARDcopy:PRINTINKSaver :HARDcopy:SAVEINKSaver :HARDcopy:SAVEFORMat :HARDcopy:ASSIGN

:HARDcopy:MODE



Description	Sets or queries whether hardcopy is set to print or save.				
Syntax	:HARDcopy:MODE { PRINT SAVE ? }				
Related Commands	:HARDcopy:START				
Parameter	<table> <tr> <td>PRINT</td> <td>Print mode</td> </tr> <tr> <td>SAVE</td> <td>Save mode</td> </tr> </table>	PRINT	Print mode	SAVE	Save mode
PRINT	Print mode				
SAVE	Save mode				
Return parameter	Returns the mode. (PRINT/SAVE)				

Example :HARDcopy:MODE PRINT
Sets hardcopy to print.

:HARDcopy:PRINTINKSaver

Set →

→ Query

Description Sets Inksaver On or Off for printing.

Syntax :HARDcopy:PRINTINKSaver { OFF | ON | ? }

Related Commands :HARDcopy:START
:HARDcopy:MODE

Parameter	ON	Inksaver ON
	OFF	Inksaver OFF

Return parameter Returns the print Ink Saver mode.(ON/OFF)

Example :HARDcopy:PRINTINKSaver ON
Sets Ink Saver to ON for printing.

Set →

→ Query

:HARDcopy:SAVEINKSaver

Description Sets Inksaver On or Off for saving screen images.

Syntax :HARDcopy:SAVEINKSaver { OFF | ON | ? }

Related Commands :HARDcopy:START
:HARDcopy:MODE

Parameter	ON	Inksaver ON
	OFF	Inksaver OFF

Return parameter Returns the screen image Ink Saver mode (ON/OFF).

Example :HARDcopy:SAVEINKSaver ON
Sets Inksaver to ON for saving screen images.

Set →

→ Query

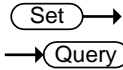
:HARDcopy:SAVEFORMat

Description Sets or queries the image save file type.

Syntax :HARDcopy:SAVEFORMat { PNG | BMP | ? }

Related Commands	:HARDcopy:START :HARDcopy:MODE	
Parameter	PNG	PNG file format
	BMP	BMP file format
Return parameter	Returns the image file format (PNG/BMP).	
Example	:HARDcopy:SAVEFORMat PNG Sets the file format to PNG.	

:HARDcopy:ASSIGN



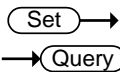
Description	Sets or queries what file type the hardcopy key has been assigned to save.	
Syntax	:HARDcopy:ASSIGN {IMAGe WAVeform SETUp ALL ?}	
Related Commands	:HARDcopy:START :HARDcopy:MODE	
Parameter	IMAGe	Save image files.
	WAVeform	Save waveforms.
	SETUp	Save the panel setup.
	ALL	Save All (image, waveform,setup)
Return parameter	Returns the file type. (IMAGE/WAVEFORM/SETUP/ALL)	
Example	:HARDcopy:ASSIGN IMAGE. Set the hardcopy key to save image files.	

Measure Commands

:MEASure:GATing.....	80
:MEASure:SOURce	80
:MEASure:METHod	81
:MEASUrement:REFLevel:PERCent:HIGH.....	81
:MEASUrement:REFLevel:PERCent:LOW	82
:MEASUrement:REFLevel:PERCent:MID	82
:MEASUrement:REFLevel:PERCent:MID2	82
:MEASure:FALL.....	83
:MEASure:FOVShoot	83
:MEASure:FPReshoot	84
:MEASure:FREQuency	84
:MEASure:NWIDth	85
:MEASure:PDUTy.....	85
:MEASure:PERiod	86
:MEASure:PWIDth	86
:MEASure:RISe.....	87
:MEASure:ROVShoot.....	87
:MEASure:RPReshoot	88
:MEASure:PPULSE.....	89
:MEASure:NPULSE	89
:MEASure:PEDGE	90
:MEASure:NEDGE	90
:MEASure:AMPlitude.....	91
:MEASure:MEAN	91
:MEASure:CMEan	92
:MEASure:HIGH	93
:MEASure:LOW	93
:MEASure:MAX	94
:MEASure:MIN	94
:MEASure:PK2PK	95
:MEASure:RMS	95
:MEASure:CRMS.....	96

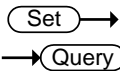
:MEASure:AREa 97
 :MEASure:CARea 97
 :MEASure:FRRDelay 98
 :MEASure:FRFDelay 98
 :MEASure:FFRDelay 99
 :MEASure:FFFDelay..... 100
 :MEASure:FLI 100
 :MEASure:LRRDelay 101
 :MEASure:LRFDelay..... 101
 :MEASure:LFRDelay..... 102
 :MEASure:LFFDelay..... 103
 :MEASure:PFLI..... 103
 :MEASure:PHase..... 104

:MEASure:GATing



Description	Sets or queries the measurement gating.	
Syntax	:MEASure:GATing { OFF SCREEn CURSor ? }	
Parameter	OFF	Full record
	SCREEn	Gating set to screen width
	CURSor	Gating between cursors
Return parameter	Returns the gating. (OFF, SCREEN, CURSOR)	
Example	:MEASure:GATing OFF Turns gating off (full record).	

:MEASure:SOURce



Description	Sets or queries the measurement source for source1 or source2.	
Syntax	:MEASure:SOURce<X> { CH1 CH2 CH3 CH4 MATH ? }	

Parameter	<X>	Source1 or source2
	CH1~CH4	Channel 1 to 4
	MATH	Math

Return parameter Returns the source (CH1, CH2, CH3, CH4, MATH)

Example :MEASure:SOURce1 CH1
Sets source1 to channel 1.

:MEASure:METHOD

Set →

→ Query

Description Sets or queries the method used to determine the High-Low measurement values.

Syntax :MEASure:METHOD { AUTO | HISTogram | MINMax | ? }

Parameter	AUTO	Set to auto.
	HISTogram	Set to the Histogram method.
	MINMax	Set to the Min-Max method.

Return parameter Returns the measurement method (AUTO, HISTOGRAM, MINMAX)

Example :MEASure:METHOD: AUTO
Set the measurement method to auto.

:MEASUrement:REFLevel:PERCent:HIGH

Set →

→ Query

Description Sets or queries the high reference level as a percentage.

Syntax :MEASUrement:REFLevel:PERCent:HIGH {<NRf> | ?}

Parameter	<NRf>	0 - 100%
-----------	-------	----------

Return parameter Returns the high reference level

Example :MEASUrement:REFLevel:PERCent:HIGH 50.1
Set the high reference level to 50.1%.

:MEASUrement:REFLevel:PERCent:LOW (Set) →
→ (Query)

Description	Sets or queries the low reference level as a percentage.	
Syntax	:MEASUrement:REFLevel:PERCent:LOW {<Nrf> ?}	
Parameter	<Nrf>	0 - 100%
Return parameter	Returns the low reference level.	
Example	:MEASUrement:REFLevel:PERCent:LOW 40.1 Set the low reference level to 40.1%.	

:MEASUrement:REFLevel:PERCent:MID (Set) →
→ (Query)

Description	Sets or queries the first mid reference level as a percentage.	
Syntax	:MEASUrement:REFLevel:PERCent:MID {<Nrf> ?}	
Parameter	<Nrf>	0 - 100%
Return parameter	Returns the mid reference level.	
Example	:MEASUrement:REFLevel:PERCent:MID 50 Set the mid reference level to 50%.	

:MEASUrement:REFLevel:PERCent:MID2 (Set) →
→ (Query)

Description	Sets or queries the second mid reference level as a percentage.	
Syntax	:MEASUrement:REFLevel:PERCent:MID2 {<Nrf> ?}	
Parameter	<Nrf>	0 - 100%
Return parameter	Returns the mid reference level of the second source.	
Example	:MEASUrement:REFLevel:PERCent:MID2 50 Set the mid reference level to 50%.	

:MEASure:FALL

→ Query

Description	Returns the fall time measurement result.	
Syntax	:MEASure:FALL{?}	
Related Commands	:MEASure:SOURce<X>	
Return parameter	<NR3>	
	Chan Off	Indicates the source channel is not activated.
Note	Before using this command, select the measurement channel. See the example below.	
Example	<pre>:MEASure:SOURce1 CH1 :MEASure:FALL?</pre> <p>Selects Channel 1 as the source, and then measures the fall time.</p>	

:MEASure:FOVShoot

→ Query

Description	Returns the fall overshoot amplitude.	
Syntax	:MEASure:FOVShoot{?}	
Related Commands	:MEASure:SOURce<X>	
Return parameter	<NR3>	Returns the fall overshoot as a percentage
	Chan Off	Indicates the source channel is not activated.
Note	Before using this command, select the measurement channel. See the example below.	

Example :MEASure:SOURce1 CH1
 :MEASure:FOVShoot?
 1.27E+0
 Selects Channel 1, and then measures the fall overshoot.

:MEASure:FPReshoot → Query

Description	Returns fall preshoot amplitude.	
Syntax	:MEASure:FPReshoot{?}	
Related Commands	:MEASure:SOURce<X>	
Returns	Returns the fall preshoot as <NR3>.	
Return parameter	<NR3>	Returns the fall preshoot as a percentage.
	Chan Off	Indicates the source channel is not activated.
Note	Before using this command, select the measurement channel. See the example below.	
Example	:MEASure:SOURce1 CH1 :MEASure:FPReshoot? Selects Channel 1, and then measures the fall preshoot.	

:MEASure:FREQuency → Query

Description	Returns the frequency value.	
Syntax	:MEASure:FREQuency{?}	
Related Commands	:MEASure:SOURce<X>	
Return parameter	<NR3>	Returns the frequency in Hz.
	Chan Off	Indicates the source channel is not activated.

Note Before using this command, select the measurement channel. See the example below.

Example :MEASure:SOURce1 CH1
 :MEASure:FREQuency?
 >1.0E+3
 Selects Channel 1, and then measures the frequency.

:MEASure:NWIDth → Query

Description Returns the first negative pulse width timing.

Syntax :MEASure:NWIDth{?}

Related Commands :MEASure:SOURce<X>

Return parameter	<NR3>	Returns the negative pulse width in seconds.
	Chan Off	Indicates the source channel is not activated.

Note Before using this command, select the measurement channel. See the example below.

Example :MEASure:SOURce1 CH1
 :MEASure:NWIDth?
 4.995E-04
 Selects Channel 1, and then measures the negative pulse width.

:MEASure:PDUTy → Query

Description Returns the positive duty cycle ratio as percentage.

Syntax :MEASure:PDUTy{?}

Related commands :MEASure:SOURce<X>

Return parameter **<NR3>** Returns the positive duty ratio.

Chan Off	Indicates the source channel is not activated.
----------	--

Note Before using this command, select the measurement channel. See the example below.

Example :MEASure:SOURce1 CH1
 :MEASure:PDUTy?
 5.000E+01
 Selects Channel 1, and then measures the positive duty cycle.

:MEASure:PERiod → Query

Description Returns the period.

Syntax :MEASure:PERiod{?}

Related Commands :MEASure:SOURce<X>

Return parameter <NR3>	Returns the period.
Chan Off	Indicates the source channel is not activated.

Note Before using this command, select the measurement channel. See the example below.

Example :MEASure:SOURce1 CH1
 :MEASure:PERiod?
 1.0E-3
 Selects Channel 1, and then measures the period.

:MEASure:PWIDth → Query

Description Returns the first positive pulse width.

Syntax :MEASure:PWIDth{?}

Related Commands :MEASure:SOURce<X>

Return parameter	<NR3> Chan Off	Returns the positive pulse width. Indicates the source channel is not activated.
Note	Before using this command, select the measurement channel. See the example below.	
Example	:MEASure:SOURce1 CH1 :MEASure:PWIDth? 5.0E-6 Selects Channel 1, and then measures the positive pulse width.	

:MEASure:RISe

→ Query

Description	Returns the first pulse rise time.	
Syntax	:MEASure:RISe{?}	
Related Commands	:MEASure:SOURce<X>	
Return parameter	<NR3> Chan Off	Returns the rise time. Indicates the source channel is not activated.
Note	Before using this command, select the measurement channel. See the example below.	
Example	:MEASure:SOURce1 CH1 :MEASure:RISe? 8.5E-6 Selects Channel 1, and then measures the rise time.	

:MEASure:ROVShoot

→ Query

Description	Returns the rising overshoot over the entire waveform in percentage.	
Syntax	:MEASure:ROVShoot{?}	

Related Commands :MEASure:SOURce<X>

Return parameter	<NR3>	Returns the overshoot.
	Chan Off	Indicates the source channel is not activated.

Note Before using this command, select the measurement channel. See the example below.

Example :MEASure:SOURce1 CH1
 :MEASure:ROVShoot?
 5.00E+00
 Selects Channel 1, and then measures the rise overshoot.

:MEASure:RPReshoot → **Query**

Description Returns rising preshoot over the entire waveform in percentage.

Syntax :MEASure:RPReshoot{?}

Related Commands :MEASure:SOURce<X>

Return parameter	<NR3>	Returns the rising preshoot.
	Chan Off	Indicates the source channel is not activated.

Note Before using this command, select the measurement channel. See the example below.

Example :MEASure:SOURce1 CH1
 :MEASure:RPReshoot?
 2.13E-2
 Selects Channel 1, and then measures the rise preshoot.

:MEASure:PPULSE

→ Query

Description	Returns the number of positive pulses.	
Syntax	:MEASure:PPULSE{?}	
Related Commands	:MEASure:SOURce<X>	
Return parameter	<NR3>	Returns the number of positive pulses.
	Chan Off	Indicates the source channel is not activated.
Note	Before using this command, select the measurement channel. See the example below.	
Example	<pre>:MEASure:SOURce1 CH1 :MEASure:PPULSE? 6.000E+00</pre> Selects Channel 1, and then measures the number of positive pulses.	

:MEASure:NPULSE

→ Query

Description	Returns the number of negative pulses.	
Syntax	:MEASure:NPULSE{?}	
Related Commands	:MEASure:SOURce<X>	
Return parameter	<NR3>	Returns the number of negative pulses.
	Chan Off	Indicates the source channel is not activated.
Note	Before using this command, select the measurement channel. See the example below.	

Example :MEASure:SOURce1 CH1
 :MEASure:NPULSE?
 4.000E+00
 Selects Channel 1, and then measures the number of negative pulses.

:MEASure:PEDGE → Query

Description Returns the number of positive edges.

Syntax :MEASure:PEDGE{?}

Related Commands :MEASure:SOURce<X>

Return parameter	<NR3>	Returns the number of positive edges.
	Chan Off	Indicates the source channel is not activated.

Note Before using this command, select the measurement channel. See the example below.

Example :MEASure:SOURce1 CH1
 :MEASure:PEDGE?
 1.100E+01
 Selects Channel 1, and then measures the number of positive edges.

:MEASure:NEDGE → Query

Description Returns the number of negative edges.

Syntax :MEASure:NEDGE{?}

Related Commands :MEASure:SOURce<X>

Return parameter	<NR3>	Returns the number of negative edges.
	Chan Off	Indicates the source channel is not activated.

Note Before using this command, select the measurement channel. See the example below.

Example :MEASure:SOURce1 CH1
 :MEASure:NEDGE?
 1.100E+01
 Selects Channel 1, and then measures the number of negative edges.

:MEASure:AMPlitude → Query

Description Returns the amplitude difference between the Vhigh-Vlow.

Syntax :MEASure:AMPlitude{?}

Related Commands :MEASure:SOURce<X>

Return parameter	<NR3>	Returns the amplitude.
	Chan Off	Indicates the source channel is not activated.

Note Before using this command, select the measurement channel. See the example below.

Example :MEASure:SOURce1 CH1
 :MEASure:AMPlitude?
 3.76E-3
 Selects Channel 1, and then measures the amplitude.

:MEASure:MEAN → Query

Description Returns the mean voltage/current of one or more full periods.

Syntax :MEASure:MEAN{?}

Related Commands :MEASure:SOURce<X>

Return parameter	<NR3> Chan Off	Returns the mean. Indicates the source channel is not activated.
Note	Before using this command, select the measurement channel. See the example below.	
Example	:MEASure:SOURce1 CH1 :MEASure:MEAN? 1.82E-3 Selects Channel 1, and then measures the mean value.	

:MEASure:CMEan → Query

Description	Returns the mean voltage/current of one full period.	
Syntax	:MEASure:CMEan{?}	
Related Commands	:MEASure:SOURce<X>	
Return parameter	<NR3> Chan Off	Returns the cyclic mean. Indicates the source channel is not activated.
Note	Before using this command, select the measurement channel. See the example below.	
Example	:MEASure:SOURce1 CH1 :MEASure:CMEan? 9.480E-01 Selects Channel 1, and then measures the mean value of the first period.	

:MEASure:HIGH

Description	Returns the global high voltage/current.	
Syntax	:MEASure:HIGH{?}	
Related Commands	:MEASure:SOURce<X>	
Return parameter	<NR3>	Returns the high value.
	Chan Off	Indicates the source channel is not activated.
Note	Before using this command, select the measurement channel. See the example below.	
Example	<pre>:MEASure:SOURce1 CH1 :MEASure:HIGH? 3.68E-3</pre> Selects Channel 1, and then measures the high voltage/current.	

:MEASure:LOW

Description	Returns the global low voltage/current.	
Syntax	:MEASure:LOW{?}	
Related Commands	:MEASure:SOURce<X>	
Return parameter	<NR3>	Returns the global low value.
	Chan Off	Indicates the source channel is not activated.
Note	Before using this command, select the measurement channel. See the example below.	

Example :MEASure:SOURce1 CH1
 :MEASure:LOW?
 1.00E-0
 Selects Channel 1, and then measures the low current/voltage.

:MEASure:MAX → Query

Description Returns the maximum amplitude.

Syntax :MEASure:MAX{?}

Related Commands :MEASure:SOURce<X>

Return parameter	<NR3>	Returns the maximum amplitude.
	Chan Off	Indicates the source channel is not activated.

Note Before using this command, select the measurement channel. See the example below.

Example :MEASure:SOURce1 CH1
 :MEASure:MAX?
 1.90E-3
 Selects Channel 1, and then measures the maximum amplitude.

:MEASure:MIN → Query

Description Returns the minimum amplitude.

Syntax :MEASure:MIN{?}

Related Commands :MEASure:SOURce<X>

Return parameter	<NR3>	Returns the minimum amplitude.
	Chan Off	Indicates the source channel is not activated.

Note Before using this command, select the measurement channel. See the example below.

Example :MEASure:SOURce1 CH1
 :MEASure:MIN?
 -8.00E-3
 Selects Channel 1, and then measures the minimum amplitude.

:MEASure:PK2PK → Query

Description Returns the peak-to-peak amplitude (difference between maximum and minimum amplitude).

Syntax :MEASure:PK2Pk{?}

Related Commands :MEASure:SOURce<X>

Return parameter	<NR3>	Returns the voltage or current peak to peak measurement.
	Chan Off	Indicates the source channel is not activated.

Note Before using this command, select the measurement channel. See the example below.

Example :MEASure:SOURce1 CH1
 :MEASure:PK2Pk?
 2.04E-1
 Selects Channel 1, and then measures the peak-to-peak amplitude.

:MEASure:RMS → Query

Description Returns the root-mean-square voltage/current of one or more full periods.

Syntax :MEASure:RMS{?}

Related Commands	:MEASure:SOURce<X>	
Return parameter	<NR3>	Returns the RMS value.
	Chan Off	Indicates the source channel is not activated.
Note	Before using this command, select the measurement channel. See the example below.	
Example	:MEASure:SOURce1 CH1 :MEASure:RMS? 1.31E-3 Selects Channel 1, and then measures the RMS voltage/current.	

:MEASure:CRMS → **Query**

Description	Returns the root-mean-square voltage/current of one full periods.	
Syntax	:MEASure:CRMS{?}	
Related Commands	:MEASure:SOURce<X>	
Return parameter	<NR3>	Returns the CRMS value.
	Chan Off	Indicates the source channel is not activated.
Note	Before using this command, select the measurement channel. See the example below.	
Example	:MEASure:SOURce1 CH1 :MEASure:CRMS? 1.31E-3 Selects Channel 1, and then measures the CRMS voltage/current.	

:MEASure:AREa

→ Query

Description	Returns the voltage/current area over one or more full periods.	
Syntax	:MEASure:AREa{?}	
Related Commands	:MEASure:SOURce<X>	
Return parameter	<NR3>	Returns the area value.
	Chan Off	Indicates the source channel is not activated.
Note	Before using this command, select the measurement channel. See the example below.	
Example	:MEASure:SOURce1 CH1 :MEASure:AREa? 1.958E-03 Selects Channel 1, and then measures the area.	

:MEASure:CARea

→ Query

Description	Returns the voltage/current area over one full period.	
Syntax	:MEASure:CARea{?}	
Related Commands	:MEASure:SOURce<X>	
Return parameter	<NR3>	Returns the area value.
	Chan Off	Indicates the source channel is not activated.
Note	Before using this command, select the measurement channel. See the example below.	

Example :MEASure:SOURce1 CH1
 :MEASure:CARea?
 1.958E-03
 Selects Channel 1, and then measures the area.

:MEASure:FRRDelay → Query

Description Returns the delay between the first rising edge of source1 and the first rising edge of source2.

Syntax :MEASure:FRRDelay{?}

Related Commands :MEASure:SOURce<X>

Return parameter	<NR3>	Returns the delay.
	Chan Off	Indicates the source channel is not activated.

Note Select the two source channels before entering this command.

Example :MEASure:SOURce1 CH1
 :MEASure:SOURce2 CH2
 :MEASure:FRRDelay?
 -4.68E-6
 Select channel 1 and 2 as source1/2, and then measure FRR.

:MEASure:FRFDelay → Query

Description Returns the delay between the first rising edge of source1 and the first falling edge of source2.

Syntax :MEASure:FRFDelay{?}

Related Commands :MEASure:SOURce<X>

Return parameter	<NR3>	Returns the delay.
------------------	-------	--------------------

	Chan Off	Indicates the source channel is not activated.
Note	Select the two source channels before entering this command.	
Example	<pre>:MEASure:SOURce1 CH1 :MEASure:SOURce2 CH2 :MEASure:FRFDelay? 3.43E-6</pre> Select channel 1 and 2 as source1/2, and then measures FRF.	

:MEASure:FFRDelay → 

Description	Returns the delay between the first falling edge of source1 and the first rising edge of source2.	
Syntax	:MEASure:FRRDelay{?}	
Related Commands	:MEASure:SOURce<X>	
Return parameter	<NR3>	Returns the delay.
	Chan Off	Indicates the source channel is not activated.
Note	Select the two source channels before entering this command.	
Example	<pre>:MEASure:SOURce1 CH1 :MEASure:SOURce2 CH2 :MEASure:FRRDelay? -8.56E-6</pre> Select channel 1 and 2 as delay source1/2, and then measure FFR.	

:MEASure:FFFDelay

→ Query

Description	Returns the delay between the first falling edge of source1 and the first falling edge of source2.	
Syntax	:MEASure:FFFDelay{?}	
Related Commands	:MEASure:SOURce<X>	
Return parameter	<NR3>	Returns the delay.
	Chan Off	Indicates the source channel is not activated.
Note	Select the two source channels before entering this command.	
Example	:MEASure:SOURce1 CH1 :MEASure:SOURce2 CH2 :MEASure:FFFDelay? -8.89E-6 Select channel 1 and 2 as delay source1/2, and then measure FFF.	

:MEASure:FLI

→ Query

Description	Returns the flicker index of times.	
Syntax	:MEASure:FLI{?}	
Related Commands	:MEASure:SOURce<X>	
Return parameter	<NR3>	Returns the flicker index of times.
	Chan Off	Indicates the source channel is not activated.
Note	Before using this command, select the measurement channel. See the example below.	

Example :MEASure:SOURce1 CH1
 :MEASure:FLI?
 2.870E-01
 Select channel 1 and then measure the flicker index of times.

:MEASure:LRRDelay → Query

Description Returns the delay between the first rising edge of source1 and the last rising edge of source2.

Syntax :MEASure:LRRDelay{?}

Related Commands :MEASure:SOURce<X>

Return parameter	<NR3>	Returns the delay.
	Chan Off	Indicates the source channel is not activated.

Note Select the two source channels before entering this command.

Example :MEASure:SOURce1 CH1
 :MEASure:SOURce2 CH2
 :MEASure:LRRDelay?
 -8.89E-6
 Select channel 1 and 2 as delay source1/2, and then measure LRR.

:MEASure:LRFDelay → Query

Description Returns the delay between the first rising edge of source1 and the last rising edge of source2.

Syntax :MEASure:LRFDelay{?}

Related Commands :MEASure:SOURce<X>

Return parameter	<NR3> Chan Off	Returns the delay. Indicates the source channel is not activated.
------------------	-------------------	--

Note Select the two source channels before entering this command.

Example :MEASure:SOURce1 CH1
 :MEASure:SOURce2 CH2
 :MEASure:LFRDelay?
 -4.99E-6
 Select channel 1 and 2 as delay source1/2, and then measure LRF.

:MEASure:LFRDelay → Query

Description Returns the delay between the first falling edge of source1 and the last rising edge of source2.

Syntax :MEASure:LFRDelay{?}

Related Commands :MEASure:SOURce<X>

Return parameter	<NR3> Chan Off	Returns the delay. Indicates the source channel is not activated.
------------------	-------------------	--

Note Select the two source channels before entering this command.

Example :MEASure:SOURce1 CH1
 :MEASure:SOURce2 CH2
 :MEASure:LFRDelay?
 -9.99E-6
 Select channel 1 and 2 as delay source1/2, and then measure LFR.

:MEASure:LFFDelay → Query

Description Returns the delay between the first falling edge of source1 and the last falling edge of source2.

Syntax :MEASure:LFFDelay{?}

Related Commands :MEASure:SOURce<X>

Return parameter	<NR3>	Returns the delay.
	Chan Off	Indicates the source channel is not activated.

Note Select the two source channels before entering this command.

Example

```
:MEASure:SOURce1 CH1
:MEASure:SOURce2 CH2
:MEASure:LFFDelay?
-9.99E-6
```

Select channel 1 and 2 as delay source1/2, and then measure LFF.

:MEASure:PFLI → Query

Description Returns the %flicker of times

Syntax :MEASure:PFLI{?}

Related Commands :MEASure:SOURce<X>

Return parameter	<NR3>	Returns the % flicker of times.
	Chan Off	Indicates the source channel is not activated.

Note Before using this command, select the measurement channel. See the example below.

Example :MEASure:SOURce1 CH1
 :MEASure:PFLI
 5.950E+01
 Select channel 1 and then measure the %flicker of times.

:MEASure:PHase

→ Query

Description Returns the phase between source 1 and source 2.

Syntax :MEASure:PHase{?}

Related Commands :MEASure:SOURce<X>

Return parameter	<NR3>	Returns the phase difference.
	Chan Off	Indicates the source channel is not activated.

Note Select the two source channels before entering this command.

Example :MEASure:SOURce1 CH1
 :MEASure:SOURce2 CH2
 :MEASure:PHase?
 4.50E+01
 Select channel 1 and 2 as phase source1/2, and then measure the phase in degrees.

Measurement Commands

:MEASUrement:MEAS<X>:SOURCE<X>	105
:MEASUrement:MEAS<X>:TYPe	106
:MEASUrement:MEAS<X>:STATE	106
:MEASUrement:MEAS<X>:VALue	107
:MEASUrement:MEAS<X>:MAXimum	108
:MEASUrement:MEAS<X>:MEAN	109
:MEASUrement:MEAS<X>:MINimum	109
:MEASUrement:MEAS<X>:STDdev	110
:MEASUrement:STATIstics:MODe	111
:MEASUrement:STATIstics:WEIghting	111
:MEASUrement:STATIstics	111

:MEASUrement:MEAS<X>:SOURCE<X>




Description	Sets or queries the measurement source for a selected automatic measurement. This is a statistics related command.	
Syntax	:MEASUrement:MEAS<X>:SOURCE<X> { CH1 CH2 CH3 CH4 MATH ? }	
Related commands	:MEASUrement:MEAS<X>:TYPe	
Parameter	MEAS<X>	The automatic measurement number from 1 to 8.
	SOURCE<X>	SOURCE1: the source for all single channel measurements.
	SOURCE<X>	SOURCE2: the source for all delay or phase measurements.
	CH1 to CH4	Channel 1, 2, 3, 4
	MATH	Math source

Return parameter	CH1 to CH4	Channel 1, 2, 3, 4
	MATH	Math source

Example :MEASUrement:MEAS1:SOURCE1?
>CH1
Returns the (first) source for measurement 1.

:MEASUrement:MEAS<X>:TYPE (Set) →
← (Query)

Description Sets or queries the measurement type for a selected automatic measurement. This is a statistics related command.

Syntax :MEASUrement:MEAS<X>:TYPE
{PK2pk | MAXimum | MINimum | AMplitude | HIGH | LOW | MEAN | CMEan | RMS | CRM s | AREa | CARea | ROVShoot | FOVShoot | RPReshoot | FPReshoot | FREQuency | PERiod | RISE | FALL | PWidth | NWidth | PDUTy | PPULSE | NPULSE | PEDGE | NEDGE | FRRDelay | FRFDelay | FFRDelay | FFFDelay | LRRDelay | LRFDelay | LFRDelay | LFFDelay | PHAse | PFLicker | FLicker?}

Related commands :MEASUrement:MEAS<X>:SOURCE<X>

Parameter	MEAS<X>	The automatic measurement number from 1 to 8.
-----------	---------	---

Return parameter Returns the measurement type

Example :MEASUrement:MEAS1:TYPE RMS
Sets measurement 1 to RMS measurement.

:MEASUrement:MEAS<X>:STATE (Set) →
← (Query)

Description Sets or queries the state of a selected measurement. This is a statistics related command.

Syntax :MEASUrement:MEAS<X>:STATE { ON | OFF | 1 | 0 | ? }

Related commands	:MEASUrement:MEAS<X>:SOURce<X> :MEASUrement:MEAS<X>:TYPE	
Parameter	MEAS<X>	The automatic measurement number from 1 to 8.
	ON/1	Turn the measurement on.
	OFF/0	Turn the measurement off.
Return parameter	0	Measurement is off.
	1	Measurement is on.
Example	:MEASUrement:MEAS1:STATE 1 Turns measurement 1 on.	

:MEASUrement:MEAS<X>:VALue → Query

Description	Returns the measurement results for the selected measurement. This is a statistics related command.	
Syntax	:MEASUrement:MEAS<X>:VALue?	
Related Commands	:MEASure:SOURce<X>	
Parameter	MEAS<X>	The automatic measurement number from 1 to 8.
Return parameter	<NR3>	Returns the measurement for the selected measurement number.
Note	The measurement source(s), measurement number, measurement type and measurement state must first be set before a measurement result can be returned.	

Example :MEASUrement:MEAS1:SOUrce1 CH1
 :MEASUrement:MEAS1:TYPe PK2PK
 :MEASUrement:MEAS1:STATE ON
 :MEASUrement:MEAS1:VALue?
 5.000E+0

Selects channel 1 as the source for measurement 1, sets measurement 1 to peak to peak measurement and then turns on the measurement. The result returns the peak to peak measurement.

:MEASUrement:MEAS<X>:MAXimum → **Query**

Description Returns the maximum measurement results for the selected measurement from the last time the statistics were reset. This is a statistics related command.

Syntax :MEASUrement:MEAS<X>:MAXimum?

Related Commands :MEASUrement:STATIstics:MODE

Parameter	MEAS<X>	The automatic measurement number from 1 to 8.
Return parameter	<NR3>	Returns the measurement for the selected measurement number.

Example :MEASUrement:MEAS3:SOUrce1 CH1
 :MEASUrement:MEAS3:TYPe PK2PK
 :MEASUrement:MEAS3:STATE ON
 :MEASUrement:STATIstics:MODE ON
 :MEASUrement:MEAS3:MAXimum?
 2.800E-02

Returns the maximum measurement result for measurement number 3.

:MEASUrement:MEAS<X>:MEAN → **Query**

Description Returns the mean measurement results for the selected measurement from the last time the statistics were reset. This is a statistics related command.

Syntax :MEASUrement:MEAS<X>:MEAN?

Related Commands :MEASUrement:STATIstics:MODE

Parameter MEAS<X> The automatic measurement number from 1 to 8.

Return parameter <NR3> Returns the measurement for the selected measurement number.

Example

```
:MEASUrement:MEAS3:SOUrce1 CH1
:MEASUrement:MEAS3:TYPe PK2PK
:MEASUrement:MEAS3:STATE ON
:MEASUrement:STATIstics:MODE ON
:MEASUrement:MEAS3:MEAN?
2.090E-02
```

Returns the mean measurement result for measurement number 3.

:MEASUrement:MEAS<X>:MINIum → **Query**

Description Returns the minimum measurement results for the selected measurement from the last time the statistics were reset. This is a statistics related command.

Syntax :MEASUrement:MEAS<X>:MINIum?

Related Commands :MEASUrement:STATIstics:MODE

Parameter MEAS<X> The automatic measurement number from 1 to 8.

Return parameter	<NR3>	Returns the measurement for the selected measurement number.
------------------	-------	--

Example

```
:MEASUrement:MEAS3:SOUrce1 CH1
:MEASUrement:MEAS3:TYPe PK2PK
:MEASUrement:MEAS3:STATE ON
:MEASUrement:STATIstics:MODE ON
:MEASUrement:MEAS3:MINIum?
1.600E-02
```

Returns the minimum measurement result for measurement number 3.

:MEASUrement:MEAS<X>:STDdev → Query

Description

Returns the standard deviation for the selected measurement from the last time the statistics were reset. This is a statistics related command.

Syntax

```
:MEASUrement:MEAS<X>:STDdev?
```

Related Commands

```
:MEASUrement:STATIstics:MODE
```

Parameter	MEAS<X>	The automatic measurement number from 1 to 8.
-----------	---------	---

Return parameter	<NR3>	Returns the measurement for the selected measurement number.
------------------	-------	--

Example

```
:MEASUrement:MEAS3:SOUrce1 CH1
:MEASUrement:MEAS3:TYPe PK2PK
:MEASUrement:MEAS3:STATE ON
:MEASUrement:STATIstics:MODE ON
:MEASUrement:MEAS3:STDdev?
1.530E-03
```

Returns the standard deviation for measurement number 3.

:MEASUREMENT:STATISTICS:MODE (Set) →
→ (Query)

Description	Puts the statics measurement results on the display or queries whether the statistics are displayed.	
Syntax	:MEASUREMENT:STATISTICS:MODE {OFF ON ?}	
Related commands	:MEASUREMENT:STATISTICS	
Parameter/ Return parameter	ON	Display the statistics on the screen.
	OFF	Remove the statistics from the screen
Example	:MEASUREMENT:STATISTICS:MODE ON Displays statistics on the screen.	

:MEASUREMENT:STATISTICS:WEIGHTING (Set) →
→ (Query)

Description	Sets and queries the number of samples (weighting) used for the statistics calculations.	
Syntax	:MEASUREMENT:STATISTICS:WEIGHTING { <NR1> ? }	
Parameter/ Return parameter	<NR1>	Number of samples (2~1000)
Example	:MEASUREMENT:STATISTICS:WEIGHTING 5 Sets the number of samples to 5.	

:MEASUREMENT:STATISTICS (Set) →

Description	Resets the statics calculations. This command will clear all the currently accumulated measurements.
Syntax	:MEASUREMENT:STATISTICS {RESET}

Reference Commands

:REF<X>:DISPlay 112
 :REF<X>:TIMebase:POSition 112
 :REF<X>:TIMebase:SCALe 113
 :REF<X>:OFFSet 113
 :REF<x>:SCALe 114

:REF<X>:DISPlay



Description Sets or queries whether a reference waveform will be shown on the display. A reference waveform must first be saved before this command can be used.

Syntax :REF<x>:DISPlay { OFF| ON| ? }

Parameter	<X>	Reference waveform 1, 2, 3, 4.
	OFF	Turns the selected reference waveform off
	ON	Turns the selected reference waveform on

Return parameter Returns the status of the selected reference waveform. (OFF, ON).

Example :REF1:DISPlay ON
 Turns on reference1 (REF 1) on the display.

:REF<X>:TIMebase:POSition



Description Sets or returns the selected reference waveform time base position.

Syntax :REF<X>:TIMebase:POSition { <NRF> | ? }

Related commands	:REF<X>:DISPlay	
Parameter	<X>	Reference waveform 1, 2, 3, 4.
	<NRf>	Horizontal co-ordinates
Return parameter	<NR3>	Returns the reference waveform position
Example	:REF1:TIMEbase:POSition -5.000E-5	
	Selects reference 1, and then sets the horizontal position to -50us.	

Set →
 → Query

:REF<X>:TIMEbase:SCALE

Description	Sets or returns the selected reference waveform time base scale.	
Syntax	:REF<X>:TIMEbase:SCALE { <NRf> ?}	
Related commands	:REF<X>:DISPlay	
Parameter	<X>	Reference waveform 1, 2, 3, 4.
	<NRf>	Horizontal scale
Return parameter	<NR3>	Returns the reference waveform scale.
Example	:REF1:TIMEbase:SCALE 5.00E-4	
	Selects reference 1, and then sets the horizontal scale to 500us/div.	

Set →
 → Query

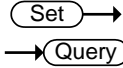
:REF<X>:OFFSet

Description	Sets or returns the selected reference waveform vertical position (offset).	
Syntax	:REF<X>:OFFSet { <NRf> ?}	
Related commands	:REF<X>:DISPlay	
Parameter	<X>	Reference waveform 1, 2, 3, 4.
	<NRf>	Vertical offset

Return parameter	<code><NR3></code>	Returns the reference waveform vertical position.
------------------	--------------------------	---

Example :REF1:OFFSet -5.000E-2
 Selects reference 1, and then sets the vertical position to -50mV/mA.

:REF<x>:SCALE



Description	Sets or returns the selected reference waveform vertical scale.	
-------------	---	--

Syntax	:REF<X>:SCALE { <NRf> ? }	
--------	-----------------------------	--

Related commands	:REF<X>:DISPlay	
------------------	-----------------	--

Parameter	<code><X></code>	Reference waveform 1, 2, 3, 4.
	<code><NRf></code>	Vertical scale

Return parameter	<code><NR3></code>	Returns the reference waveform vertical scale.
------------------	--------------------------	--

Example :REF1:SCALE 5.000E-2
 Selects reference 1, and then sets the vertical scale to 50mV | mA/div.

Run Command

:RUN

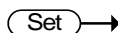


Description	The run command allows the oscilloscope to continuously make acquisitions (equivalent to pressing the Run key on the front panel).
-------------	--

Syntax	:RUN
--------	------

Stop Command

:STOP



Description	The stop command stops the oscilloscope making further acquisitions (equivalent to pressing the Stop key on the front panel).
-------------	---

Syntax	:STOP
--------	-------

Single Command

:SINGle



Description	The single command allows the oscilloscope to capture a single acquisition when trigger conditions have been fulfilled (equivalent to pressing the Single key on the front panel).
-------------	--

Syntax	:SINGle
--------	---------

Force Command

:FORCe

Set →

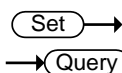
Description	The Force command forces an acquisition (equivalent to pressing the Force-Trig key on the front panel).
-------------	---

Syntax	:FORCe
--------	--------

Timebase Commands

:TIMebase:EXPand	117
:TIMebase:POSition	117
:TIMebase:SCALe.....	118
:TIMebase:MODE	118
:TIMebase:WINDow:POSition	118
:TIMebase:WINDow:SCALe	119

:TIMebase:EXPand

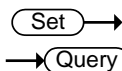


Description Sets or queries the horizontal expansion mode.

Syntax :TIMebase:EXPand {CENTer|TRIGger|?}

Parameter/Return parameter	CENTer	Expand from the center of the display.
	TRIGger	Expand from the trigger point.

Example :TIMebase:EXPand TRIGger
 Sets the expansion point to the trigger point.



:TIMebase:POSition

Description Sets or queries the horizontal position.

Syntax :TIMebase:POSition {<NRf> | ?}

Parameter	<NRf>	Horizontal position
Return parameter	<NR3>	Returns the horizontal position

Example :TIMebase:POSition 5.00E-4
 Sets the horizontal position as 500us.

Set →

→ Query

:TIMEbase:SCALE

Description	Sets or queries the horizontal scale.	
Syntax	:TIMEbase:SCALE {<NRf> ?}	
Parameter	<NRf>	Horizontal scale
Return parameter	<NR3>	Returns the horizontal scale.
Example	:TIMEbase:SCALE 5.00E-2 Sets the horizontal scale to 50ms/div.	

Set →

→ Query

:TIMEbase:MODE

Description	Sets or queries the time base mode. The time base mode determines the display view window on the scope.	
Syntax	:TIMEbase:MODE {MAIN WINDOW XY ?}	
Parameter	MAIN	Sets the time base mode to the main screen.
	WINDOW	Sets the time base mode to the zoom window.
	XY	Sets the time base mode to the XY display.
Return parameter	Returns the time base mode (MAIN, WINDOW, XY)	
Example	:TIMEbase:MODE MAIN Sets the time base mode to the main mode.	

Set →

→ Query

:TIMEbase:WINDOW:POSITION

Description	Sets or queries the zoom horizontal position.	
Syntax	:TIMEbase:WINDOW:POSITION {<NRf> ?}	

Related commands	:TIMebase:MODE	
Parameter	<NRf>	Horizontal position for zoom window
Return parameter	<NR3>	Returns the zoom horizontal position.
Example	:TIMebase:WINDow:POSition 2.0E-3 Sets the zoom horizontal position as 20ms.	

:TIMebase:WINDow:SCALE

Set →

→ Query

Description	Sets or queries the zoom horizontal scale.	
Note	If the oscilloscope is under “ZOOM” mode, the main timebase function will be disabled and cannot be modified.	
Syntax	:TIMebase:WINDow:SCALE {<NRf> ?}	
Related commands	:TIMebase:MODE	
Parameter	<NRf>	Zoom horizontal scale. The range will depend on the time base.
Return parameter	<NR3>	Returns the zoom horizontal scale.
Example	:TIMebase:WINDow:SCALE 2.0E-3 Sets the zoom horizontal scale to 2ms.	

Trigger Commands

:TRIGger:FREQuency.....	122
:TRIGger:TYPe	122
:TRIGger:SOURce	123
:TRIGger:COUPle.....	123
:TRIGger:NREJ	123
:TRIGger:MODe.....	124
:TRIGger:HOLDoff.....	124
:TRIGger:LEVel.....	124
:TRIGger:HLEVel	125
:TRIGger:LLEVel.....	125
:TRIGger:EDGE:SLOP.....	126
:TRIGger:DElay:SLOP	126
:TRIGger:DElay:TYPe.....	127
:TRIGger:DElay:TIme.....	127
:TRIGger:DElay:EVENt.....	127
:TRIGger:DElay:LEVel	128
:TRIGger:PULSEWidth:POLarity	128
:TRIGger:RUNT:POLarity	129
:TRIGger:RUNT:WHEn.....	129
:TRIGger:RUNT:TIme.....	130
:TRIGger:RISEFall:SLOP.....	130
:TRIGger:RISEFall:WHEn	130
:TRIGger:RISEFall:TIme	131
:TRIGger:VIDeo:TYPe	131
:TRIGger:VIDeo:FIELd	132
:TRIGger:VIDeo:LINE	132
:TRIGger:VIDeo:POLarity	133
:TRIGger:PULSe:WHEn	133
:TRIGger:PULSe:TIme	134
:TRIGger:TIMEOut:WHEn.....	134
:TRIGger:TIMEOut:TImER	135
:TRIGger:ALTErnate	135

:TRIGger:STATe	135
:TRIGger:EXTERnal:PROBe:TYPe	136
:TRIGger:EXTERnal:PROBe:RATio	137
:TRIGger:BUS:TYPe	137
:TRIGger:BUS:THReshold:CH<x>	137
:TRIGger:BUS:B1:I2C:CONDition	138
:TRIGger:BUS:B1:I2C:ADDReSS:MODe	139
:TRIGger:BUS:B1:I2C:ADDReSS:TYPe	139
:TRIGger:BUS:B1:I2C:ADDReSS:VALue	140
:TRIGger:BUS:B1:I2C:ADDReSS:DIRection	140
:TRIGger:BUS:B1:I2C:DATA:SIze	141
:TRIGger:BUS:B1:I2C:DATA:VALue	141
:TRIGger:BUS:B1:UART:CONDition	142
:TRIGger:BUS:B1:UART:RX:DATA:SIze	143
:TRIGger:BUS:B1:UART:RX:DATA:VALue	143
:TRIGger:BUS:B1:UART:TX:DATA:SIze	144
:TRIGger:BUS:B1:UART:TX:DATA:VALue	144
:TRIGger:BUS:B1:SPI:CONDition	145
:TRIGger:BUS:B1:SPI:DATA:SIze	146
:TRIGger:BUS:B1:SPI:DATA:MISO:VALue	146
:TRIGger:BUS:B1:SPI:DATA:MOSI:VALue	147
:TRIGger:BUS:B1:CAN:CONDition	148
:TRIGger:BUS:B1:CAN:FRAMeType	148
:TRIGger:BUS:B1:CAN:IDentifier:MODe	149
:TRIGger:BUS:B1:CAN:IDentifier:VALue	149
:TRIGger:BUS:B1:CAN:IDentifier:DIRection	150
:TRIGger:BUS:B1:CAN:DATA:QUALifier	150
:TRIGger:BUS:B1:CAN:DATA:SIze	151
:TRIGger:BUS:B1:CAN:DATA:VALue	152
:TRIGger:BUS:B1:LIN:CONDition	152
:TRIGger:BUS:B1:LIN:DATA:QUALifier	153
:TRIGger:BUS:B1:LIN:DATA:SIze	154
:TRIGger:BUS:B1:LIN:DATA:VALue	154
:TRIGger:BUS:B1:LIN:ERRTYPe	155
:TRIGger:BUS:B1:LIN:IDentifier:VALue	156

:TRIGger:FREQuency

→ **Query**

Description	Queries the trigger frequency.
Syntax	:TRIGger:FREQuency{?}
Return parameter	<NR3> Returns the trigger frequency.
Example	:TRIGger:FREQuency? 1.032E+3 Returns the trigger frequency.

Set →

:TRIGger:TYPE

→ **Query**

Description	Sets or queries the trigger type.
Syntax	:TRIGger:TYPE {EDGE DELay PULSEwidth VIDeo RUNT RISEFall LOGic BUS TIMEOut ? }
Parameter	EDGE Edge trigger DELay Delay trigger PULSEwidth Pulse width trigger VIDeo Video trigger RUNT Runt trigger RISEFall Rise and fall trigger LOGic Logic trigger BUS Bus trigger TIMEOut Timeout trigger
Return parameter	Returns the trigger type.
Example	:TRIGger:TYPE EDGE Sets the trigger type to edge.

:TRIGger:SOURce
 →
 →

Description	Sets or queries the trigger source.	
Syntax	:TRIGger:SOURce { CH1 CH2 CH3 CH4 EXT LINe ? }	
Parameter	CH1 to CH4	Channel 1 to channel 4
	EXT	External source
	LINe	AC Line
Return parameter	Returns the trigger source.	
Example	:TRIGger:SOURce CH1 Sets the trigger source to channel 1.	

:TRIGger:COUPlE
 →
 →

Description	Sets or queries the trigger coupling.	
Note	Applicable for edge and delay triggers only.	
Syntax	:TRIGger:COUPlE {AC DC HF LF ?}	
Parameter	AC	AC mode
	DC	DC mode
	HF	High frequency rejection
	LF	Low frequency rejection
Return parameter	Returns the trigger coupling.	
Example	:TRIGger:COUPlE AC Sets the trigger coupling to AC.	

:TRIGger:NREJ
 →
 →

Description	Sets or queries noise rejection status.	
Syntax	:TRIGger:NREJ {OFF ON ?}	

Parameter	OFF	Turns noise rejection off
	ON	Turns noise rejection on

Return parameter Returns the noise rejection status (ON, OFF).

Example :TRIGger:NREJ ON
Turns noise rejection on.

Set →

→ Query

:TRIGger:MODE

Description Sets or queries the trigger mode.

Syntax :TRIGger:MODE {AUTo | NORMal | ?}

Parameter	AUTo	Auto trigger (Untriggered roll)
	NORMal	Normal trigger

Return parameter Returns the trigger mode.

Example :TRIGger:MODE NORMal
Sets the trigger mode to normal.

Set →

→ Query

:TRIGger:HOLDoff

Description Sets or queries the holdoff time.

Syntax :TRIGger:HOLDoff {<NRf> | ?}

Parameter	<NRf>	Holdoff time
-----------	-------	--------------

Return parameter <NR3> Returns the trigger holdoff time.

Example :TRIGger:HOLDoff 1.00E-8
Sets the trigger holdoff time to 10ns.

Set →

→ Query

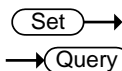
:TRIGger:LEVel

Description Sets or queries the level.

Note Not applicable to Pulse Runt and Rise & Fall triggers.

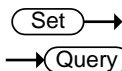
Syntax	:TRIGger:LEVel {TTL ECL SETTO50 <NRf> ?}	
Related commands	:TRIGger:TYPe	
Parameter	<NRf>	Trigger level value.
	TTL	Sets the trigger level to TTL.
	ECL	Sets the trigger level to ECL.
	SETTO50	Sets the trigger level to the User level (50% by default).
Return parameter	<NR3>	Returns the trigger level.
Example1	:TRIGger:LEVel TTL Sets the trigger to TTL.	
Example2	:TRIGger:LEVel 3.30E-1 Sets the trigger level to 330mV/mA.	

:TRIGger:HLEVel



Description	Sets or queries the high trigger level.	
Note	Applicable for Rise and Fall/Pulse Runt triggers.	
Syntax	:TRIGger:HLEVel {<NRf> ?}	
Related commands	:TRIGger:TYPe	
Parameter	<NRf>	High level value.
Return parameter	<NR3>	Returns the trigger high level.
Example	:TRIGger:HLEVel 3.30E-1 Sets the trigger high level to 330mV/mA.	

:TRIGger:LLEVel



Description	Sets or queries the low trigger level.	
Note	Applicable for Rise and Fall/Pulse Runt triggers.	

Syntax	:TRIGger:LLEVel {<NRf> ?}	
Related commands	:TRIGger:TYPe	
Parameter	<NRf>	Low level value.
Return parameter	<NR3>	Returns the trigger low level.
Example	:TRIGger:LLEVel -3.30E-3 Sets the trigger low level to -330mV/mA.	

Set →

→ Query

:TRIGger:EDGE:SLOP

Description	Sets or queries the trigger slope.	
Syntax	:TRIGger:EDGE:SLOP {RISe FALL EITHer ? }	
Related commands	:TRIGger:TYPe	
Parameter	RISe	Rising slope
	FALL	Falling slope
	EITHer	Either rising or falling slope
Return parameter	Returns the trigger slope.	
Example	:TRIGger:EDGE:SLOP FALL Sets the trigger slope to falling.	

Set →

→ Query

:TRIGger:DELAy:SLOP

Description	Sets or queries the trigger slope for the delay trigger.	
Syntax	:TRIGger:DELAy:SLOP {RISe FALL EITHer ? }	
Related commands	:TRIGger:TYPe	
Parameter	RISe	Rising slope
	FALL	Falling slope
	EITHer	Either rising or falling slope

Return parameter Returns the trigger slope.

Example :TRIGger:DElay:SLOP FALL
Sets the trigger slope to falling.

Set →

→ Query

:TRIGger:DElay:TYPe

Description Sets or queries the trigger delay type.

Syntax :TRIGger:DElay:TYPe {TImE | EVEnt | ?}

Related commands :TRIGger:TYPe

Parameter	TImE	Sets the delay type to time.
	EVEnt	Sets the delay type to event.

Return parameter Returns the trigger delay type.

Example :TRIGger:DElay:TYPe TImE
Sets the delay type to time delay.

Set →

→ Query

:TRIGger:DElay:TImE

Description Sets or queries the delay time value.

Syntax :TRIGger:DElay:TImE {<NRf> | ?}

Related commands :TRIGger:DElay:TYPe

Parameter	<NRf>	Delay time (1.00E-8~1.00E+1)
-----------	-------	------------------------------

Return parameter	<NR3>	Returns the delay time.
------------------	-------	-------------------------

Example :TRIGger:DElay:TImE 1.00E-6
Sets the delay time to 1us.

Set →

→ Query

:TRIGger:DElay:EVEnt

Description Sets or queries the number of events for the event delay trigger.

Syntax	:TRIGger:DElay:EVENT {<NR1> ?}	
Related commands	:TRIGger:DElay:TYPe	
Parameter	<NR1>	1~65535 events
Return parameter	<NR1>	Returns the number of events.
Example	:TRIGger:DElay:EVENT 2 Sets the number of events to 2.	

Set →
 → Query

:TRIGger:DElay:LEVel

Description	Sets or queries the trigger delay level.	
Syntax	:TRIGger:DElay:LEVel {<NRf> ?}	
Parameter	<NRf>	Delay trigger level
Return parameter	<NR3>	Returns the delay trigger.
Example	:TRIGger:DElay:LEVel 5.00E-3 Sets the delay trigger level to 5mV/mA.	

Set →
 → Query

:TRIGger:PULSEWidth:POLarity


Description	Sets or queries the pulse width trigger polarity.	
Syntax	:TRIGger:PULSEWidth:POLarity {POSitive NEGative ?}	
Related commands	:TRIGger:TYPe	
Parameter	POSitive	Positive polarity
	NEGative	Negative polarity
Return parameter	Returns the pulse width polarity.	
Example	:TRIGger:PULSEWidth:POLarity POSitive Sets the pulse width polarity to positive.	

:TRIGger:RUNT:POLarity

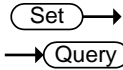



Description	Sets or queries the Pulse Runt trigger polarity.	
Syntax	:TRIGger:RUNT:POLarity { POSitive NEGative EITher ? }	
Related commands	:TRIGger:TYPE	
Parameter	POSitive	Positive polarity
	NEGative	Negative polarity
	EITher	Positive or negative polarity
Return parameter	Returns the pulse runt trigger polarity.	
Example	:TRIGger:RUNT:POLarity POSitive Sets the Pulse Runt trigger polarity to positive.	

:TRIGger:RUNT:WHEn

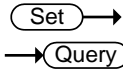



Description	Sets or queries the Pulse Runt trigger conditions.	
Syntax	:TRIGger:RUNT:WHEn { MOREthan LESSthan EQual UNEQual ? }	
Related commands	:TRIGger:TYPE :TRIGger:RUNT:TIME	
Parameter	MOREthan	>
	LESSthan	<
	EQual	=
	UNEQual	≠
Return parameter	Returns the pulse runt trigger condition.	
Example	:TRIGger:RUNT:WHEn UNEQual Sets the Pulse Runt trigger condition to unequal (≠).	



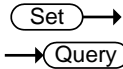
:TRIGger:RUNT:TIME

Description	Sets or queries the Pulse Runt trigger time.	
Syntax	:TRIGger:RUNT:TIME {<NRf> ? }	
Related commands	:TRIGger:TYPe :TRIGger:RUNT:WHEn	
Parameter	<NRf>	Pulse runt time (4nS to 10S)
Return Parameter	<NR3>	Returns the runt time in seconds.
Example	:TRIGger:RUNT:TIME 4.00E-5 Sets the runt time to 40.0uS.	



:TRIGger:RISEFall:SLOP

Description	Sets or queries the Rise & Fall slope.	
Syntax	:TRIGger:RISEFall:SLOP {RISe FALL EITHer ? }	
Parameter	RISe	Rising slope
	FALL	Falling slope
	EITHer	Either rising or falling slope
Return parameter	Returns the rise & fall slope.	
Example	:TRIGger:RISEFall:SLOP RISe Sets the Rise & Fall slope to rising.	



:TRIGger:RISEFall:WHEn

Description	Sets or queries the rise/fall trigger conditions.	
Syntax	:TRIGger:RISEFall:WHEn {MOREthan LESSthan EQual UNEQual ? }	
Related commands	:TRIGger:TYPe :TRIGger:RISEFall:TIME	
Parameter	MOREthan	>

LESSthan	<
Equal	=
UNEQUAL	≠

Return parameter Returns the rise/fall trigger condition.

Example :TRIGger:RISEFall:WHEn UNEQUAL
Sets the Rise and Fall trigger condition to unequal (≠).

:TRIGger:RISEFall:TIME (Set) →
→ (Query)

Description Sets or queries the Rise and Fall time.

Syntax :TRIGger:RISEFall:TIME {<NRf> | ? }

Related commands :TRIGger:TYPe
:TRIGger:RISEFall:WHEn

Parameter	<NRf>	Rise and Fall time (4nS to 10S)
Return Parameter	<NR3>	Returns the rise and fall time in seconds.

Example :TRIGger:RISEFall:TIME 4.00E-5
Sets the trigger rise & fall to 40.0us.

:TRIGger:VIDeo:TYPe (Set) →
→ (Query)

Description Sets or queries the video trigger type.

Syntax :TRIGger:VIDeo:TYPe {NTSC | PAL | SECam | EDTV480P | EDTV576P | HDTV720P | HDTV1080I | HDTV1080P | ? }

Related commands :TRIGger:TYPe

Parameter	NTSC	NTSC
	PAL	PAL
	SECam	SECAM

EDTV480P	Extra definition TV 480P
EDTV576P	Extra definition TV 576P
HDTV720P	High definition TV 720P
HDTV1080I	High definition TV 1080i
HDTV1080P	High definition TV 1080P

Return parameter Returns the video trigger type.

Example :TRIGger:VIDeo:TYPe NTSC
Sets the video trigger to NTSC.

Set →

→ Query

:TRIGger:VIDeo:FIELD

Description Sets or queries the video trigger field.

Syntax :TRIGger:VIDeo:FIELD { FIELD1 | FIELD2 | ALLFields | ALLLines | ? }

Related commands :TRIGger:TYPe

Parameter	FIELD1	Trigger on field 1
	FIELD2	Trigger on field 2
	ALLFields	Trigger on all fields
	ALLLines	Trigger on all lines

Return parameter Returns the video trigger field.

Example :TRIGger:VIDeo:FIELD ALLFields
Sets the video trigger to trigger on all fields.

Set →

→ Query

:TRIGger:VIDeo:LINE

Description Sets or queries the video trigger line.

Syntax :TRIGger:VIDeo:LINE {<NR1> | ?}

Related commands :TRIGger:TYPe

Parameter	<NR1>	Video line
Return parameter	<NR3>	Returns the video trigger line.
Example	:TRIGger:VIDeo:LINE 1 Sets the video trigger to line 1.	

:TRIGger:VIDeo:POLarity




Description	Sets or queries the video trigger polarity.	
Syntax	:TRIGger:VIDeo:POLarity { POSitive NEGative ? }	
Related commands	:TRIGger:TYPE	
Parameter	POSitive	Positive polarity
	NEGative	Negative polarity
Return parameter	Returns the video trigger polarity.	
Example	:TRIGger:VIDeo:POLarity POSitive Sets the video trigger polarity to positive.	

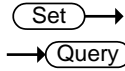
:TRIGger:PULSe:WHEn




Description	Sets or queries the pulse width trigger conditions.	
Syntax	:TRIGger:PULSe:WHEn { MOREthan LESSthan EQual UNEQual ? }	
Related commands	:TRIGger:TYPE :TRIGger:PULSe:TIME	
Parameter	MORE than	>
	LESSthan	<
	EQual	=
	UNEQual	≠

Return parameter Returns the pulse width trigger conditions.

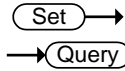
Example :TRIGger:PULSe:WHEn UNEQual
 Sets the trigger pulse width conditions to not equal to (≠).



:TRIGger:PULSe:TIME

Description	Sets or queries the pulse width time.	
Syntax	:TRIGger:PULSe:TIME {<NRf> ?}	
Related commands	:TRIGger:TYPe :TRIGger:PULSe:WHEn	
Parameter	<NRf>	Pulse width time (4ns~10s)
Return parameter	<NR3>	Returns the pulse width time in seconds.


Example :TRIGger:PULSe:TIME 4.00E-5
 Sets the trigger pulse width to 40.0us.

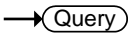


:TRIGger:TIMEOut:WHEn

Description	Sets or queries the timeout trigger condition.	
Syntax	:TRIGger:TIMEOut:WHEn {HIGH LOW EITHer ?}	
Related commands	:TRIGger:TIMEOut:TIMER	
Parameter	HIGH	Signal is high.
	LOW	Signal is low.
	EITHer	Signal is high or low.
Return parameter	Returns the timeout condition (HIGH, LOW, EITHER).	

Example1 :TRIGger:TIMEOut:WHEn LOW
 Sets the timeout condition to low.

:TRIGger:TIMEOut:TIMER




Description	Sets or returns timeout trigger time.
Syntax	:TRIGger:TIMEOut:TIMER {<NRf> ? }
Related commands	:TRIGger:TIMEOut:WHEn
Parameter	<NRf> Timeout time. (4nS to 10S).
Return parameter	Returns the timeout time as <NR3>.
Example	:TRIGger:TIMEOut:TIMER? 8.960e-05

:TRIGger:ALTErnate




Description	Sets alternating between source triggers on or off or queries its state.
Syntax	:TRIGger:ALTErnate {OFF ON ?}
Parameter	OFF Alternate off ON Alternate on
Return parameter	Returns the Alternate trigger status (ON, OFF).
Example	:TRIGger:ALTErnate ON Turns on alternating between source triggers.

:TRIGger:STATe


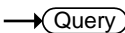

Description	Returns the current state of the triggering system.
Syntax	:TRIGger:STATe?
Return parameter	*ARMED Indicates that the oscilloscope is acquiring pretrigger information.

*AUTO	Indicates that the oscilloscope is in the automatic mode and acquires data even in the absence of a trigger.
*READY	Indicates that all pretrigger information has been acquired and that the oscilloscope is ready to accept a trigger.
*SAVE	Indicates that the oscilloscope is in save mode and is not acquiring data.
*TRIGGER	Indicates that the oscilloscope triggered and is acquiring the post trigger information.

Example :TRIGger:STATe?
 AUTO
 The trigger is in auto mode.

Set →
 → Query


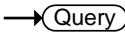
Description	Sets or queries the external probe type.	
Syntax	:TRIGger:EXTERnal:PROBE:TYPE { VOLTage CURRent ? }	
Related commands	:TRIGger:EXTERnal:PROBE:RATio	
Parameter	VOLTage	Voltage
	CURRent	Current
Return parameter	Returns the probe type.	
Example	:TRIGger:EXTERnal:PROBE:TYPE? CURRENT	

:TRIGger:EXTERnal:PROBe:RATio



Description	Sets or queries the external probe ratio (attenuation).	
Syntax	:TRIGger:EXTERnal:PROBe:RATio {<NRf> ?}	
Related commands	:TRIGger:EXTERnal:PROBe:TYPe	
Parameter	<NRf>	External probe attenuation factor.
Return parameter	<NR3>	Returns the probe attenuation factor.
Example	:TRIGger:EXTERnal:PROBe:RATio? 5.000000e+01	

:TRIGger:BUS:TYPe 

Description	Returns the current bus type.	
Syntax	:TRIGger:BUS:TYPe?	
Return parameter	12C	I ² C mode
	SPI	SPI mode
	UART	UART mode
	CAN	CAN mode
	LIN	LIN mode
Example	:TRIGger:BUS:TYPe? UART	

:TRIGger:BUS:THReshold:CH<x>



Description	Sets or queries the threshold level for the selected channel.	
Syntax	:TRIGger:BUS:THReshold:CH<X> {<NR3> ?}	
	<X>	CH1 ~ CH4

	<NR3>	Threshold level
--	-------	-----------------

Return Parameter	<NR3>	Returns the threshold level
------------------	-------	-----------------------------

Example :TRIGger:BUS:THReshold:CH1 1
Sets the CH1 threshold to 1V.

:TRIGger:BUS:B1:I2C:CONDition




Description	Sets or queries the I ² C trigger conditions.	
-------------	--	--

Syntax	:TRIGger:BUS:B1:I2C:CONDition {START STOP REPEATstart ACKMISS ADDRess DATA ADDRANDDATA ? }	
--------	---	--

Parameter	START	Set Start as the I ² C trigger condition.
	STOP	Set Stop as the I ² C trigger condition.
	REPEATstart	Set Repeat of Start as the I ² C trigger condition.
	ACKMISS	Set Missing Acknowledgement as the I ² C trigger condition.
	ADDRess	Set Address as the I ² C trigger condition.
	DATA	Set Data as the I ² C trigger condition.
	ADDRANDDATA	Set Address and Data as the I ² C trigger condition.

Return parameter	Returns the I ² C bus trigger condition.	
------------------	---	--

Example :TRIGger:BUS:B1:I2C:CONDition ADDRess
Set Address as the I²C trigger condition.

:TRIGger:BUS:B1:I2C:ADDRess:MODE



Description	Sets or queries the I ² C addressing mode (7 or 10 bits).	
Syntax	:TRIGger:BUS:B1:I2C:ADDRess:MODE {ADDR7 ADDR10 ? }	
Related commands	:TRIGger:BUS:B1:I2C:CONDition	
Parameter	ADDR7	7 bit addressing
	ADDR10	10 bit addressing
Return Parameter	0	7 bit addressing
	1	10 bit addressing
Example	:TRIGger:BUS:B1:I2C:ADDRess:MODE? 0 The addressing mode is currenty set to 7 bits.	

:TRIGger:BUS:B1:I2C:ADDRess:TYPE



Description	Sets the I ² C bus address type, or queries what the setting is.	
Syntax	:TRIGger:BUS:B1:I2C:ADDRess:TYPE {GENERALcall STARTbyte HSmode EEPROM CBUS ? }	
Related commands	:TRIGger:BUS:B1:I2C:CONDition	
Parameter	GENERALcall	Set a general call address (0000 000 0).
	STARTbyte	Set a start byte address. (0000 000 1)
	HSmode	Set a high-speed mode address. (0000 1xx x)
	EEPROM	Set an EEPROM address. (1010 xxx x)
	CBUS	Set a CBUS address. (0000 001 x)

Return Parameter Returns the address type

Example :TRIGger:BUS:B1:I2C:ADDRes:TYPe?
CBUS

:TRIGger:BUS:B1:I2C:ADDRes:VALue

Set →

← Query

Description Sets or queries the I²C bus address value when the I²C bus is set to trigger on Address or Address/Data.

Syntax :TRIGger:BUS:B1:I2C:ADDRes:VALue {<string> | ? }

Related commands :TRIGger:BUS:B1:I2C:ADDRes:MODE

Parameter	<sting>	7/10 characters, must be enclosed in double quotes, "string". x = don't care 1 = binary 1 0 = binary 0
-----------	---------	---

Return Parameter Returns the address value.

Example1 :TRIGger:BUS:B1:I2C:ADDRes:MODE ADDR7
:TRIGger:BUS:B1:I2C:ADDRes:VALue "xxx0101"
Sets the address to XXX0101

Example 2 :TRIGger:BUS:B1:I2C:ADDRes:VALue?
XXX0101

Set →

← Query

Description Sets or queries the address bit as read write or don't care.

Note This setting only applies when the I²C trigger is set to trigger on Address or Address/Data

Syntax :TRIGger:BUS:B1:I2C:ADDRes:DIRectioN { READ | WRITE | NOCARE | ? }

Related commands	:TRIGger:BUS:B1:I2C:CONDition	
Parameter	READ	Set read as the data direction.
	WRITE	Set write as the data direction.
	NOCARE	Set either as the data direction.
Return Parameter	Returns the direction (READ, WRITE, NOCARE).	
Example	:TRIGger:BUS:B1:I2C:ADDReSS:DIRectiON READ Sets the direction to READ.	

:TRIGger:BUS:B1:I2C:DATA:SIZE



Description	Sets or queries the data size in bytes for the I ² C bus.	
Note	This setting only applies when the I ² C trigger is set to trigger on Data or Address/Data	
Syntax	:TRIGger:BUS:B1:I2C:DATA:SIZE {<NR1> ? }	
Related commands	:TRIGger:BUS:B1:I2C:CONDition	
Parameter	<NR1>	Number of data bytes (1 to 5).
Return parameter	<NR1>	Returns the number of bytes.
Example	:TRIGger:BUS:B1:I2C:DATA:SIZE 3 Sets the number of bytes to 3.	

:TRIGger:BUS:B1:I2C:DATA:VALue



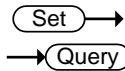
Description	Sets or queries the triggering data value for the I ² C bus when the I ² C bus is set to trigger on Data or Address/Data.	
Syntax	:TRIGger:BUS:B1:I2C:DATA:VALue {<string> ? }	
Related commands	:TRIGger:BUS:B1:I2C:DATA:SIZE	

Parameter	<string>	The number of characters in the string depends on the data size setting. The string must be enclosed in double quotes, "string". x = don't care 1 = binary 1 0 = binary 0
-----------	----------	--

Return Parameter Returns the data value.

Example1 :TRIGger:BUS:B1:I2C:DATA:SIZE 1
:TRIGger:BUS:B1:I2C:DATA:VALue "1x1x0101"
Sets the value to XXX0101

Example 2 :TRIGger:BUS:B1:I2C:DATA:VALue?
1X1X0101



:TRIGger:BUS:B1:UART:CONDition

Description Sets or queries the UART triggering condition.

Syntax :TRIGger:BUS:B1:UART:CONDition { RXSTArt | RXDATA | RXENDPacket | TXSTArt | TXDATA | TXENDPacket | TXPARIttyerr | RXPARIttyerr | ? }

Parameter	RXSTArt	Set trigger on the RX Start Bit.
	RXDATA	Set trigger on RX Data.
	RXENDPacket	Set trigger on the RX End of Packet condition.
	RXPARIttyerr	Set trigger on RX Parity error condition.
	TXSTArt	Set trigger on the TX Start Bit.
	TXDATA	Set trigger on TX Data.
	TXENDPacket	Set trigger on the TX End of Packet condition.
	TXPARIttyerr	Set trigger on TX Parity error condition.

Return Parameter Returns the triggering condition.

Example :TRIGger:BUS:B1:UART:CONDition TXDATA
Sets the UART bus to trigger on Tx Data.

:TRIGger:BUS:B1:UART:RX:DATA:SIZE (Set) →
→ (Query)

Description Sets or queries the number of bytes for UART data.

Note This setting only applies when the UART trigger is set to trigger on Rx Data

Syntax :TRIGger:BUS:B1:UART:RX:DATA:SIZE {<NR1> | ?}

Related commands :TRIGger:BUS:B1:UART:CONDition

Parameter <NR1> Number of bytes (1 to 10).

Return parameter <NR1> Returns the number of bytes.

Example :TRIGger:BUS:B1:UART:RX:DATA:SIZE 5
Sets the number of bytes to 5.

:TRIGger:BUS:B1:UART:RX:DATA:VALue (Set) →
→ (Query)

Description Sets or queries the triggering data value for the UART bus when the bus is set to trigger on Rx Data.

Syntax :TRIGger:BUS:B1:UART:RX:DATA:VALue {<string> | ?}

Related commands :TRIGger:BUS:B1:UART:RX:DATA:SIZE

Parameter <sting> The number of characters in the string depends on the data size setting. The string must be enclosed in double quotes, "string".
x = don't care
1 = binary 1
0 = binary 0

Return Parameter Returns the data value.

Example 1 :TRIGger:BUS:B1:UART:CONDition RXDATA
 :TRIGger:BUS:B1:UART:RX:DATA:SIZE 1
 :TRIGger:BUS:B1:UART:RX:DATA:VALue "1x1x0101"
 Sets the value to 1x1x0101

Example 2 :TRIGger:BUS:B1:UART:RX:DATA:VALue?
 1X1X0101

:TRIGger:BUS:B1:UART:TX:DATA:SIZE  

Description Sets or queries the number of bytes for UART data.

Note This setting only applies when the UART trigger is set to trigger on Tx Data

Syntax :TRIGger:BUS:B1:UART:TX:DATA:SIZE {<NR1> | ?}

Related commands :TRIGger:BUS:B1:UART:CONDition

Parameter <NR1> Number of bytes (1 to 10).

Return parameter <NR1> Returns the number of bytes.

Example :TRIGger:BUS:B1:UART:TX:DATA:SIZE 5
 Sets the number of bytes to 5.

:TRIGger:BUS:B1:UART:TX:DATA:VALue  

Description Sets or queries the triggering data value for the UART bus when the bus is set to trigger on Tx Data.

Syntax :TRIGger:BUS:B1:UART:TX:DATA:VALue {<string> | ?}

Related commands :TRIGger:BUS:B1:UART:TX:DATA:SIZE

Parameter	<sting>	The number of characters in the string depends on the data size setting. The string must be enclosed in double quotes, "string". x = don't care 1 = binary 1 0 = binary 0
-----------	---------	--

Return Parameter Returns the data value.

Example1 :TRIGger:BUS:B1:UART:CONDition TXDATA
:TRIGger:BUS:B1:UART:TX:DATA:SIZE 1
:TRIGger:BUS:B1:UART:TX:DATA:VALue "1x1x0101"
Sets the value to 1x1x0101

Example 2 :TRIGger:BUS:B1:UART:TX:DATA:VALue?
1X1X0101

:TRIGger:BUS:B1:SPI:CONDition  

Description Sets or queries the SPI triggering condition.

Syntax :TRIGger:BUS:B1:SPI:CONDition {SS | MISO | MOSI | MISOMOSI | ? }

Parameter	SS	Set to trigger on the Slave Selection condition.
	MISO	Set to trigger on the Master-In Slave-Out condition.
	MOSI	Set to trigger on the Master-Out Slave-In condition.
	MISOMOSI	Set to trigger on the Master-In Slave-Out and Master-Out Slave-In conditions.

Return Parameter Returns the triggering condition.

Example :TRIGger:BUS:B1:SPI:CONDition MISO
Sets the SPI bus to trigger on MISO.

Set →
 → Query

:TRIGger:BUS:B1:SPI:DATA:SIZE

Description	Sets or queries the number of words for SPI data.	
Note	This setting only applies when the SPI trigger is set to trigger on MISO, MOSI or MISO/MOSI	
Syntax	:TRIGger:BUS:B1:SPI:DATA:SIZE {<NR1> ?}	
Related commands	:TRIGger:BUS:B1:SPI:CONDition	
Parameter	<NR1>	Number of words (1 to 32).
Return parameter	<NR1>	Returns the number of words.
Example	:TRIGger:BUS:B1:SPI:DATA:SIZE 10 Sets the number of words to 10.	

Set →
 → Query

:TRIGger:BUS:B1:SPI:DATA:MISO:VALue

Description	Sets or queries the triggering data value for the SPI bus when the bus is set to trigger on MISO or MISO/MOSI.	
Syntax	:TRIGger:BUS:B1:SPI:DATA:MISO:VALue {<string> ?}	
Related commands	:TRIGger:BUS:B1:SPI:DATA:SIZE	
Parameter	<sting>	The number of characters in the string depends on the data size setting. The string must be enclosed in double quotes, "string". x = don't care 1 = binary 1 0 = binary 0
Return Parameter	Returns the data value.	

Example1 :TRIGger:BUS:B1:SPI:CONDition MISO
 :TRIGger:BUS:B1:SPI:DATA:SIZE 2
 :TRIGger:BUS:B1:SPI:DATA:MISO:VALue "1x1x0101"
 Sets the value to 1x1x0101

Example 2 :TRIGger:BUS:B1:SPI:DATA:MISO:VALue?
 1X1X0101

:TRIGger:BUS:B1:SPI:DATA:MOSI:VALue 

Description Sets or queries the triggering data value for the SPI bus when the bus is set to trigger on MOSI or MISO/MOSI.

Syntax :TRIGger:BUS:B1:SPI:DATA:MOSI:VALue {<string> | ? }

Related commands :TRIGger:BUS:B1:SPI:DATA:SIZE

Parameter	<code><string></code>	The number of characters in the string depends on the data size setting. The string must be enclosed in double quotes, "string". x = don't care 1 = binary 1 0 = binary 0
-----------	-----------------------------	--

Return Parameter Returns the data value.

Example1 :TRIGger:BUS:B1:SPI:CONDition MOSI
 :TRIGger:BUS:B1:SPI:DATA:SIZE 2
 :TRIGger:BUS:B1:SPI:DATA:MOSI:VALue "1x1x0101"
 Sets the value to 1x1x0101

Example2 :TRIGger:BUS:B1:SPI:DATA:MOSI:VALue?
 1X1X0101

:TRIGger:BUS:B1:CAN:CONDition (Set) →
→ (Query)

Description	Sets or returns the CAN trigger condition.	
Syntax	:TRIGger:BUS:B1:CAN:CONDition {SOF FRAMeType IDentifier DATA IDANDDATA EOF ACKMISS STUFFERR ?}	
Parameter/ Return parameter	SOF	Triggers on a start of frame
	FRAMeType	Triggers on the type of frame
	IDentifier	Triggers on a matching identifier
	DATA	Triggers on matching data
	IDANDDATA	Triggers on matching identifier and data field
	EOF	Triggers on the end of frame
	ACKMISS	Triggers on a missing acknowledge
	STUFFERR	Triggers on a bit stuffing error

Example1 :TRIGger:BUS:B1:CAN:CONDition SOF
Triggers on a start of frame.

Example2 :TRIGger:BUS:B1:CAN:CONDition?
>SOF

:TRIGger:BUS:B1:CAN:FRAMeType (Set) →
→ (Query)

Description	Sets or returns the frame type for a CAN FRAMeType trigger.	
Syntax	:TRIGger:BUS:B1:CAN:FRAMeType {DATA REMote ERRor OVERLoad ?}	
Parameter/ Return parameter	DATA	Sets the frame type to data frame
	REMote	Sets the frame type to remote frame
	ERRor	Sets the frame type to error frame
	OVERLoad	Sets the frame type to overload

Example :TRIGger:BUS:B1:CAN:FRAMeType DATA
Sets the frame type to DATA.

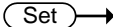
:TRIGger:BUS:B1:CAN:IDentifier:MODE  

Description Sets or returns the CAN identifier mode for the bus.

Syntax :TRIGger:BUS:B1:CAN:IDentifier:MODE {STANDard|EXTended|?}

Parameter/ Return parameter	STANDard	Standard addressing mode
	EXTended	Extended addressing mode

Example :TRIGger:BUS:B1:CAN:IDentifier:MODE?>STANDARD
Returns the addressing mode.

:TRIGger:BUS:B1:CAN:IDentifier:VALue  

Description Sets or returns the identifier string used for the CAN trigger.
Note: Only applicable when the trigger condition is set to ID or IDANDDATA.

Syntax :TRIGger:BUS:B1:CAN:IDentifier:VALue {<string>|?}

Related Commands :TRIGger:BUS:B1:CAN:IDentifier:MODE

Parameter/ Return parameter	<string>	The size of the string depends on the data size setting. The string must be enclosed in double quotes, "string". String contents: x = don't care 1 = binary 1 0 = binary 0
--------------------------------	----------	--

```
Example      :TRIGger:BUS:B1:CAN:CONDition ID
              :TRIGger:BUS:B1:CAN:IDentifier:MODE STANDARD
              :TRIGger:BUS:B1:CAN:IDentifier:VALue
              "01100X1X01X"
              :TRIGger:BUS:B1:CAN:IDentifier:VALue?
              >01100X1X01X
```

Set →

```
:TRIGger:BUS:B1:CAN:IDentifier:DIRection → Query
```

Description Sets or queries the address bit as read, write or don't care.

Syntax :TRIGger:BUS:B1:CAN:IDentifier:DIRection {READ|WRITE|NOCARE|?}

Parameter/ Return parameter	READ	Sets read as the data direction
	WRITE	Sets write as the data direction
	NOCARE	Sets either as the data direction

Example1 :TRIGger:BUS:B1:CAN:IDentifier:DIRection? >WRITE

Example2 :TRIGger:BUS:B1:CAN:IDentifier:DIRection READ
:TRIGger:BUS:B1:CAN:IDentifier:DIRection? > READ

Set →

```
:TRIGger:BUS:B1:CAN:DATA:QUALifier → Query
```

Description Sets or returns the CAN data qualifier.
Note: Only applicable when the triggering condition is set to DATA or IDANDDATA.

Syntax :TRIGger:BUS:B1:CAN:DATA:QUALifier {LESSthan|MOREthan|Equal|UNEQual|LESSEQual|MOREEQual|?}

Parameter/ Return parameter	LESSthan	Triggers when the data is less than the qualifier value.
--	----------	--

MOREthan	Triggers when the data is greater than the qualifier value.
EQual	Triggers when the data is equal to the qualifier value.
UNEQual	Triggers when the data is not equal to the qualifier value.
LESSEQual	Triggers when the data is less than or equal to the qualifier value.
MOREEQual	Triggers when the data is more than or equal to the qualifier value.

Example :TRIGger:BUS:B1:CAN:DATA:QUALifier?
 >EQUAL
 :TRIGger:BUS:B1:CAN:DATA:QUALifier MOREthan
 :TRIGger:BUS:B1:CAN:DATA:QUALifier?
 >MOREthan

:TRIGger:BUS:B1:CAN:DATA:SIZE

Set →

← Query

Description Sets or returns the length of the data string in bytes for a CAN trigger.

Note: Only applicable when the condition is set to DATA or IDANDDATA.

Syntax :TRIGger:BUS:B1:CAN:DATA:SIZE {<NR1>|?}

Parameter/ Return parameter	<NR1>	1~8 (bytes)
--------------------------------	-------	-------------

Example :TRIGger:BUS:B1:CAN:DATA:SIZE?
 >1
 :TRIGger:BUS:B1:CAN:DATA:SIZE 2
 :TRIGger:BUS:B1:CAN:DATA:SIZE?
 >2

Set →

→ Query

:TRIGger:BUS:B1:CAN:DATA:VALue

Description Sets or returns the binary data string to be used for a CAN trigger.

Note: Only applicable when the condition is set to DATA or IDANDDATA.

Related Commands :TRIGger:BUS:B1:CAN:DATA:SIZE

Syntax :TRIGger:BUS:B1:CAN:DATA:VALue {<string>|?}

Parameter/ Return parameter	<string>	The size of the string depends on the data size setting. The string must be enclosed in double quotes, "string". String contents: x = don't care 1 = binary 1 0 = binary 0
--	-----------------------	--

Example :TRIGger:BUS:B1:CAN:DATA:SIZE 1
 :TRIGger:BUS:B1:CAN:DATA:VALue "01010X1X"
 :TRIGger:BUS:B1:CAN:DATA:VALue?
 >01010X1X

Set →

→ Query

:TRIGger:BUS:B1:LIN:CONDition

Description Sets or returns the LIN trigger condition.

Syntax :TRIGger:BUS:B1:LIN:CONDition
 {SYNCField|IDentifier|DATA|IDANDDATA|WAKEup|SLEEP|ERRor|?}

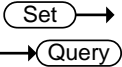
Parameter/ Return parameter	SYNCField	Sets the LIN trigger condition to the sync field.
	IDentifier	Sets the LIN trigger condition to identifier field.

DATA	Sets the LIN trigger condition to the data field.
IDANDDATA	Sets the LIN trigger condition to identifier and data field
WAKEup	Sets the LIN trigger condition to wake up.
SLEEP	Sets the LIN trigger condition to sleep.
ERRor	Sets the LIN trigger condition to error.

Example

```
:TRIGger:BUS:B1:LIN:CONDition?
>IDANDDATA

:TRIGger:BUS:B1:LIN:CONDition DATA
:TRIGger:BUS:B1:LIN:CONDition?
>DATA
```

:TRIGger:BUS:B1:LIN:DATA:QUALifier 

Description

Sets or returns the LIN data qualifier.
 Note: Only applicable when the trigger condition is set to DATA or IDANDDATA.

Syntax

```
:TRIGger:BUS:B1:LIN:DATA:QUALifier
{LESSthan|MOREthan|EQual|UNEQual|LESSEQual|M
OREEQual|?}
```

Parameter/ Return parameter	LESSthan	Triggers when the data is less than the qualifier value.
	MOREthan	Triggers when the data is greater than the qualifier value.
	EQual	Triggers when the data is equal to the qualifier value.
	UNEQual	Triggers when the data is not equal to the qualifier value.
	LESSEQual	Triggers when the data is less than or equal to the qualifier value.

MOREEqual Triggers when the data is more than or equal to the qualifier value.

Example :TRIGger:BUS:B1:LIN:DATA:QUALifier?
>EQUAL
:TRIGger:BUS:B1:LIN:DATA:QUALifier MOREthan
:TRIGger:BUS:B1:LIN:DATA:QUALifier?
>MORETHAN

:TRIGger:BUS:B1:LIN:DATA:SIZE (Set) →
← (Query)

Description Sets or returns the length of the data string in bytes for the LIN trigger.
Note: Only applicable when the condition is set to DATA or IDANDDATA.

Syntax :TRIGger:BUS:B1:LIN:DATA:SIZE {<NR1>|?}

Parameter/ Return parameter <NR1> 1~8 (bytes)

Example :TRIGger:BUS:B1:LIN:DATA:SIZE?
>1
:TRIGger:BUS:B1:LIN:DATA:SIZE 2
:TRIGger:BUS:B1:LIN:DATA:SIZE?
>2

:TRIGger:BUS:B1:LIN:DATA:VALue (Set) →
← (Query)

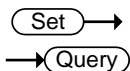
Description Sets or returns the binary data string to be used for the LIN trigger.
Note: Only applicable when the condition is set to DATA or IDANDDATA.

Related Commands :TRIGger:BUS:B1:LIN:DATA:SIZE

Syntax :TRIGger:BUS:B1:LIN:DATA:VALue {<string>|?}

Parameter/ Return parameter	<string>	The size of the string depends on the data size setting. The string must be enclosed in double quotes, "string". String contents: x = don't care 1 = binary 1 0 = binary 0
--------------------------------	----------	--

Example :TRIGger:BUS:B1:LIN:DATA:SIZE 1
 :TRIGger:BUS:B1:LIN:DATA:VALue "01010X1X"
 :TRIGger:BUS:B1:LIN:DATA:VALue?
 >01010X1X



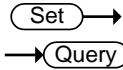
:TRIGger:BUS:B1:LIN:ERRTYPE

Description Sets or returns the error type be used for the LIN trigger.

Syntax :TRIGger:BUS:B1:LIN:ERRTYPE
 {SYNC|PARItY|CHecksum|?}

Parameter/ Return parameter	SYNC	Sets the LIN error type to SYNC.
	PARItY	Sets the LIN error type to parity.
	CHecksum	Sets the LIN error type to checksum.

Example :TRIGger:BUS:B1:LIN:ERRTYPE?
 >SYNC
 :TRIGger:BUS:B1:LIN:ERRTYPE CHECKSUM
 :TRIGger:BUS:B1:LIN:ERRTYPE?
 >CHECKSUM




:TRIGger:BUS:B1:LIN:IDentifier:VALue

Description	Sets or returns the identifier string to be used for the LIN trigger. Note: Only applicable when the condition is set to ID or IDANDDATA.	
Syntax	:TRIGger:BUS:B1:LIN:IDentifier:VALue {<string> ?}	
Parameter/ Return parameter	<string>	The size of the string depends on the data size setting. The string must be enclosed in double quotes, "string". String contents: x = don't care 1 = binary 1 0 = binary 0
Example	:TRIGger:BUS:B1:LIN:CONDition ID :TRIGger:BUS:B1:LIN:IDentifier:VALue "00X1X01X" :TRIGger:BUS:B1:LIN:IDentifier:VALue? >01100X1X01X	

System Commands

:SYSTem:LOCK	157
:SYSTem:ERRor.....	157

:SYSTem:LOCK  

Description	Turns the panel lock on off.	
Syntax	:SYSTem:LOCK {OFF ON ? }	
Parameter	OFF	System lock off
	ON	System lock on
Return parameter	Returns the status of the panel lock (ON, OFF).	
Example	:SYSTem:LOCK ON Turns the panel lock on.	

:SYSTem:ERRor  

Description	Queries the error queue. See the appendix on page 285 for details.	
Syntax	:SYSTem:ERRor?	
Return parameter	Returns the last message in the error queue.	
Example	:SYSTem:ERRor? +0, "No error."	

Save/Recall Commands

:RECALL:SETUp	158
:RECALL:WAVEform	158
:SAVE:IMAGe	159
:SAVE:IMAGe:FILEFormat.....	159
:SAVE:IMAGe:INKSaver.....	160
:SAVE:SETUp	160
:SAVE:WAVEform.....	161
:SAVE:WAVEform:FILEFormat	162

:RECALL:SETUp



Description	Recalls setup settings from memory or USB.	
Syntax	:RECALL:SETUp {S1~S20 <file path>("Disk:/xxx.SET", "USB:/xxx.SET")}	
Parameter	S1~S20	Recall Set1~Set20
	<file path>	Recall a file from the DSO internal files system or from a USB flash drive.
Example	<p>:RECALL:SETUp S1 Recalls setup setting S1 from memory.</p> <p>:RECALL:SETUp "Disk:/DS0001.SET" Recall the setup setting DS0001.SET from the internal memory.</p>	

:RECALL:WAVEform



Description	Recalls a waveform from wave1~wave20 or from file to REF1~4.
Note	Detail CSV files cannot be recalled.
Syntax	:RECALL:WAVEform {W<n> <file path> ("Disk:/xxx.LSF", "USB:/xxx.LSF")}, REF<X>

Parameter	n	1~20 (Wave1~wave20)
	<file page>	Filename in file path. Example: “Disk:/xxx.LSF”, “USB:/xxx.LSF”, “Disk:/xxx.CSV”, “USB:/xxx.CSV”
	<X>	1,2,3,4 (REF1, REF2, REF3, REF4)

Example :RECALL:WAVEform W1, REF1
 Recalls the waveform stored in Wave1 to reference 1.

:SAVe:IMAGe



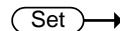
Description Saves a screen image to the assigned file path with a specified filename.

Syntax :SAVe:IMAGe {<file path> (“Disk:/xxx.PNG”, “USB:/xxx.BMP”)}

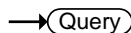
Related commands :SAVe:IMAGe:FILEFormat
 :SAVe:IMAGe:INKSaver

Parameter	xxx.PNG or xxx.BMP	File name (8 characters max)
-----------	--------------------	------------------------------

Example :SAVe:IMAGe “Disk:/pic1.PNG”
 Saves a screen image named pic1.png to the root directory (Disk:/) of the scope.
 :SAVe:IMAGe “USB:/pic1.BMP”
 Saves a screen image named pic1.bmp to the root directory of the external USB flash disk.



:SAVe:IMAGe:FILEFormat



Description Sets the file format for image.

Syntax :SAVe:IMAGe:FILEFormat {PNG | BMP | ?}

Related commands :SAVe:IMAGe
 :SAVe:IMAGe:INKSaver

Parameter	PNG	Sets the file format to PNG
	BMP	Sets the file format to BMP

Return parameter Returns the file format (PNG, BMP).

Example :SAVE:IMAGe:FILEFormat PNG
Sets the image file format to PNG.

Set →

→ Query

:SAVE:IMAGe:INKSaver

Description Turns Ink Saver on or off.

Syntax :SAVE:IMAGe:INKSaver {OFF | ON |?}

Related commands :SAVE:IMAGe
:SAVE:IMAGe:FILEFormat

Parameter	OFF	Turns Inksaver off.
	ON	Turns Inksaver on.

Return parameter Returns Ink Saver status (ON, OFF).

Example :SAVE:IMAGe:INKSaver ON
Turns Ink Saver on.

Set →

:SAVE:SETUp

Description Saves the current setup to internal memory (Set1~Set20) or the designated file path.

Syntax :SAVE:SETUp {<file path> ("Disk:/xxx.SET", "USB:/xxx.SET) | S1~S20}

Parameter	S1~S20	Saves the setup to Set1~Set20
	File path	Saves the setup to disk to the specified file path.

Example :SAVe:SETUp S1
 Saves the current setup to Set1 in internal memory.
 :SAVe:SETUp "Disk:/DS0001.SET"
 Saves the current setup to DS0001.SET in the root directory of the internal memory.

:SAVe:WAVEform Set →

Description Saves a waveform to internal memory or to a designated file path.

Related commands :SAVe:WAVEform:FILEFormat

Syntax :SAVe:WAVEform {CH1~REF4, REF<X> } | {CH1~REF4, W1~W20} | {CH1~ALL, file path}

Parameter	CH1~REF4, <X> W1~W20 ALL File path	CH1~CH4, Math, D0~D15, REF1~4 1,2,3,4 (REF1, REF2, REF3, REF4) Wave1~Wave20 All the displayed waveforms on screen. Saves the waveform(s) to disk or USB to the specified file path. (LSF or CSV, but note that detail CSV can't be recalled to the scope.)
------------------	--	--

Example 1 :SAVe:WAVEform CH1, REF2
 Saves the channel1 waveform to REF2.

Example 2 :SAVe:WAVEform:FILEFormat LSF
 :SAVe:WAVEform ALL, "Disk:/ALL001"
 Sets the file format to LSF. A folder named "ALL001" is created and saves all displayed waveforms to the "ALL001" directory in the LSF format.

Example 3 :SAVE:WAVEform:FILEFormat FCSV
 :SAVE:WAVEform ALL, "Disk:/ALL002"
 Sets the file format to FCSV(fast CSV format). It then saves the all channel's waveforms to the root directory (Disk:/) of the internal flash disk in the CSV format (with the filename ALL002.CSV).

Example 4 :SAVE:WAVEform:FILEFormat LSF
 :SAVE:WAVEform CH2, "Disk:/DS0003.LSF"
 Save the channel 2's waveform to the root directory (Disk:/) of the internal flash disk in the LSF format with DS0003.LSF as the filename.

:SAVE:WAVEform:FILEFormat
 →
 ←

Description	Sets the waveform savefile format.	
Syntax	:SAVE:WAVEform:FILEFormat {LSF DCSV FCSV ?}	
Parameter	LSF	Sets the file format to the MDO-2000EG/2000EX's internal file format, LSF. (xxx.LSF) (no support LA)
	DCSV	Sets the file format to detail CSV. (xxx.CSV)
	FCSV	Sets the file format to fast CSV. (xxx.CSV)
Return parameter	Returns the file format (LSF , DCSV, FCSV).	

Example :SAVE:WAVEform:FILEFormat LSF
 Sets the file format to LSF.

Ethernet Command

:ETHERnet:DHCP

Set →

→ Query

Description	Sets or queries the DHCP settings.	
Syntax	:ETHERnet:DHCP { OFF ON ? }	
Parameter	ON	Turns DHCP on.
	OFF	Turns DHCP off.
Example	:ETHERnet:DHCP ON Turns DHCP on.	

Time Command

:DATE

Set →

Description	Sets the system date and time.	
Syntax	:DATE {<string>}	
Parameter	<string>	“YYYYMMDDhhmmss”
		Where: YYYY: year MM: month DD: day hh: hour mm: minute ss: second
Example	:DATE “20140802142830” Sets the time and date as: Year: 2014, Month: 08, Day: 02, Hour: 14 (2PM), Minute: 28, Second: 30.	

Bus Decode Commands

:BUS1	165
:BUS1:STATE	165
:BUS1:TYPE	165
:BUS1:INPut	166
:BUS1:I2C:ADDReSS:RWINClude	166
:BUS1:I2C:SCLK:SOURce	167
:BUS1:I2C:SDA:SOURce	167
:BUS1:UART:BITRate.....	167
:BUS1:UART:DATABits.....	168
:BUS1:UART:PARItY.....	168
:BUS1:UART:PACKet	168
:BUS1:UART:EOFPAcket	169
:BUS1:UART:TX:SOURce	169
:BUS1:UART:RX:SOURce	169
:BUS1:SPI:SCLK:POLARity	170
:BUS1:SPI:SS:POLARity.....	170
:BUS1:SPI:WORDSize	170
:BUS1:SPI:BITORder	171
:BUS1:SPI:SCLK:SOURce	171
:BUS1:SPI:SS:SOURce.....	171
:BUS1:SPI:MOSI:SOURce	172
:BUS1:SPI:MISO:SOURce	172
:BUS1:DISPlay:FORMAt	172
:LISTer:DATA	173
:BUS1:CAN:SOURce.....	173
:BUS1:CAN:PROBe.....	173
:BUS1:CAN:SAMPLEpoint.....	174
:BUS1:CAN:BITRate	174
:BUS1:LIN:BITRate	175
:BUS1:LIN:IDFORmat	175
:BUS1:LIN:POLARity	175
:BUS1:LIN:SAMPLEpoint	176

:BUS1:LIN:SOURce176
 :BUS1:LIN:STANDard176

:BUS1

→ Query

Description	Returns the supported BUS types.
Syntax	:BUS1?
Return Parameter	Returns the supported bus types.
Example	BUS1? I2C,SPI,UART,CAN,LIN

Set →

:BUS1:STATE

→ Query

Description	Sets or queries the state of the bus.
Syntax	:BUS1:STATE { OFF ON ? }
Related commands	:BUS1:TYPe
Parameter/Return parameter	OFF Turns the bus off. ON Turns the bus on.
Example	:BUS1:STATE ON Turns the bus on.

Set →

:BUS1:TYPe

→ Query

Description	Sets or queries the type of bus.
Syntax	:BUS1:TYPe { UART I2C SPI PARAllel CAN LIN ? }
Related commands	:BUS1:STATE
Parameter/Return parameter	UART Sets the bus to UART mode. I2C Sets the bus to I ² C mode.

	SPI	Sets the bus to SPI mode.
	PARAllel	Sets the bus to parallel mode.
	CAN	Sets the bus to CAN mode.
	LIN	Sets the bus to LIN mode.

Example :BUS1:TYPE SPI
Sets the bus to SPI mode.

:BUS1:INPut Set →
← Query

Description	Sets or returns the bus source.	
Syntax	:BUS1:INPut {ANALog DIGital ?}	
Parameter/Return parameter	ANALog	Sets the bus source as analog inputs.
	DIGital	Sets the bus source as digital inputs.

Example1 :BUS1:INPut?
>ANALOG

:BUS1:I2C:ADDRes:RWINClude Set →
← Query

Description	Sets or queries whether the read/write bit is included in the I ² C address.	
Syntax	:BUS1:I2C:ADDRes:RWINClude { OFF ON ? }	
Related commands	:BUS1:STATE	
Parameter	OFF	The R/W bit is not included.
	ON	The R/W bit is included.
Return parameter	0	The R/W bit is not included.
	1	The R/W bit is included.

Example :BUS1:I2C:ADDRes:RWINClude ON
Includes the R/W bit in the I²C address.

Set →
 → Query

:BUS1:I2C:SCLK:SOURce

Description Sets or queries which channel is used for the I²C SCLK source.

Syntax :BUS1:I2C:SCLK:SOURce { CH1 | CH2 | CH3 | CH4 |? }

Parameter/Return parameter CH1 to CH4 Analog channels 1 ~ 4.

Example :BUS1:I2C:SCLK:SOURce CH1
 Sets channel 1 as the SCLK source.

Set →
 → Query

:BUS1:I2C:SDA:SOURce

Description Sets or queries which channel is used for the I²C SDA source.

Syntax :BUS1:I2C:SDA:SOURce { CH1 | CH2 | CH3 | CH4 |? }

Parameter/Return parameter CH1 to CH4 Analog channels 1 ~ 4.

Example :BUS1:I2C:SDA:SOURce CH1
 Sets channel 1 as the SDA source.

Set →
 → Query

:BUS1:UART:BITRate

Description Sets or queries the UART bit rate.

Syntax :BUS1:UART:BITRate {<NR1> |? }

Parameter/Return parameter <NR1> UART bit rate in bps

Example :BUS1:UART:BITRate?
 >2400
 :BUS1:UART:BITRate 50
 :BUS1:UART:BITRate?
 >50

Set →

→ Query

:BUS1:UART:DATABits

Description	Sets or queries the number UART data for bus 1.	
Syntax	:BUS1:UART:DATABits { 5 6 7 8 9 ? }	
Parameter/Return parameter	5	5 data bits in the UART frame.
	6	6 data bits in the UART frame.
	7	7 data bits in the UART frame.
	8	8 data bits in the UART frame.

Example :BUS1:UART:DATABits 7
Sets the UART frame to 7 bits.

Set →

→ Query

:BUS1:UART:PARItY

Description	Sets or queries the UART bus parity.	
Syntax	:BUS1:UART:PARItY { <NR1> ? }	
Parameter/Return parameter	<NR1>	0: None
		1: Odd parity
		2: Even parity

Example :BUS1:UART:PARItY 1
Sets the parity to odd.


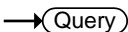
Set →

→ Query

:BUS1:UART:PACKEt

Description	Sets or queries the UART packet setting.	
Syntax	:BUS1:UART:PACKEt {<NR1> ? }	
Parameter/Return parameter	<NR1>	0: Off
		1: On

Example :BUS1:UART:PACKEt 1
Turns UART packets on.

:BUS1:UART:EOFPAcket



Description Sets or queries the EOF character for the UART packet setting.

Syntax :BUS1:UART:EOFPAcket <NR1>

Parameter/Return parameter	<NR1>	0: NULL 1: LF (line feed) 2: CR (carriage return) 3: SP (space character) 4: FF
-----------------------------------	-------	---

Example :BUS1:UART:EOFPAcket 2
Sets the OEF character to CR.

:BUS1:UART:TX:SOURce



Description Sets or queries which channel is used for the UART Tx source.

Syntax :BUS1:UART:TX:SOURce { OFF | CH1 | CH2 | CH3 | CH4 | ? }

Parameter/Return parameter	OFF	Off, no Tx source
	CH1 to CH4	Analog channels CH1 to CH4

Example :BUS1:UART:TX:SOURce CH1
Sets channel 1 as the Tx source.

:BUS1:UART:RX:SOURce



Description Sets or queries which channel is used for the UART Rx source.

Syntax :BUS1:UART:RX:SOURce { OFF | CH1 | CH2 | CH3 | CH4 | ? }

Parameter/Return	OFF	Off, no Rx source
-------------------------	-----	-------------------

parameter CH1 to CH4 Analog channels CH1 to CH4

Example :BUS1:UART:RX:SOURce CH1
Sets channel 1 as the Rx source.

:BUS1:SPI:SCLK:POLARity (Set) →
→ (Query)

Description Sets or queries the polarity of the SCLK line for the SPI bus.

Syntax :BUS1:SPI:SCLK:POLARity { FALL | RISE | ? }

Parameter/Return parameter FALL Sets the polarity to falling edge.

:BUS1:SPI:SS:POLARity (Set) →
→ (Query)

Description Sets or queries the polarity of the SS line for the SPI bus.

Syntax :BUS1:SPI:SS:POLARity { LOW | HIGH | ? }

Parameter/Return parameter LOW Active low polarity
HIGH Active high polarity

Example :BUS1:SPI:SS:POLARity LOW
Sets the SS line to active low.

:BUS1:SPI:WORDSize (Set) →
→ (Query)

Description Sets the number of bits per word for the SPI bus.

Syntax :BUS1:SPI:WORDSize { <NR1> | ? }

Parameter/Return parameter <NR1> Bits per word (4-32)

Example :BUS1:SPI:WORDSize 4
Sets the word size to 4 bits per word.





:BUS1:SPI:BITORder

Description	Sets or queries the bit order for the SPI bus.	
Syntax	:BUS1:SPI:BITORder {<NR1> ? }	
Parameter/Return parameter	<NR1>	0: MSB bit first 1: LSB bit first
Example	:BUS1:SPI:BITORder? 0 The bit order is currently set as MSB bit first.	





:BUS1:SPI:SCLK:SOURce

Description	Sets or queries which channel is used for the SPI SCLK source.	
Syntax	:BUS1:SPI:SCLK:SOURce { CH1 CH2 CH3 CH4 ? }	
Parameter/Return parameter	CH1 to CH4	Analog channels CH1 to CH4
Example	:BUS1:SPI:SCLK:SOURce CH1 Sets channel 1 as the SPI SCLK source.	





:BUS1:SPI:SS:SOURce

Description	Sets or queries which channel is used for the SPI SS source.	
Syntax	:BUS1:SPI:SS:SOURce { CH1 CH2 CH3 CH4 D0 ? }	
Parameter/Return parameter	CH1 to CH4	Analog channels CH1 to CH4
Example	:BUS1:SPI:SS:SOURce CH1 Sets channel 1 as the SPI SS source.	

Set →
 → Query

:BUS1:SPI:MOSI:SOURce

Description	Sets or queries which channel is used for the SPI MOSI source.	
Syntax	:BUS1:SPI:MOSI:SOURce { OFF CH1 CH2 CH3 CH4 ? }	
Parameter/Return parameter	CH1 to CH4	Analog channels CH1 to CH4
	OFF	No MOSI source.
Example	:BUS1:SPI:MOSI:SOURce CH1 Sets channel 1 as the SPI MOSI source.	

Set →
 → Query

:BUS1:SPI:MISO:SOURce

Description	Sets or queries which channel is used for the SPI MISO source.	
Syntax	:BUS1:SPI:MISO:SOURce { OFF CH1 CH2 CH3 CH4 ? }	
Parameter/Return parameter	CH1 to CH4	Analog channels CH1 to CH4
	OFF	No MISO source.
Example	:BUS1:SPI:MISO:SOURce CH1 Sets channel CH1 as the SPI MISO source.	

Set →
 → Query

:BUS1:DISplay:FORMAt

Description	Sets or queries the display format for the bus, either binary or hexadecimal.	
Syntax	:BUS1:DISplay:FORMAt { BINary HEXadecimal ASCII ? }	
Parameter/Return parameter	BINary	Binary format
	HEXadecimal	Hexadecimal format

Example : BUS1:DISplay:FORMAt BINary
Sets the display format to binary.

:LISTer:DATA → Query

Description Returns the Event Table data as a binary block data.

Syntax :LISTer:DATA?

Return Parameter Returns the event table as binary block data. The binary block data contains comma separated data with new lines at the end of each row.

Set →

:BUS1:CAN:SOURce → Query

Description Sets or returns the CAN input source.

Syntax :BUS1:CAN:SOURce { CH1 | CH2 | CH3 | CH4 | ? }

Parameter/Return parameter CH1 ~ CH4 Analog channel source

Example :BUS1:CAN:SOURCE?
>CH1
Returns the CAN source.

Set →

:BUS1:CAN:PROBe → Query

Description Sets or returns the signal type of the CAN bus.

Syntax :BUS1:CAN:PROBe { CANH | CANL | TX | RX | ? }

Parameter/Return parameter	CANH	CAN-High
	CANL	CAN-Low
	TX	Transmit
	RX	Receive

```
Example      :BUS1:CAN:PROBe?
              >CANH
              :BUS1:CAN:PROBe CANL
              :BUS1:CAN:PROBe?
              >CANL
```

:BUS1:CAN:SAMPLEpoint → Query

Description Returns the sample point of the CAN bus.

Syntax :BUS1:CAN:SAMPLEpoint?

Return Parameter Returns the sample point of the CAN bus as a percentage of the bit time.

```
Example      :BUS1:CAN:SAMPLEpoint?
              50
              Returns the sample point as a percentage.
```

Set →

:BUS1:CAN:BITRate → Query

Description Sets or returns the bit rate of the CAN bus.

Syntax :BUS1:CAN:BITRate
 {RATE10K|RATE20K|RATE50K|RATE125K|RATE250K|
 RATE500K|RATE800K|RATE1M | <NR1> | ?}

Parameter/Return parameter	RATE10K	10 kbps
	RATE20K	20 kbps
	RATE50K	50 kbps
	RATE125K	125 kbps
	RATE250K	250 kbps
	RATE500K	500 kbps
	RATE800K	800 kbps
	RATE1M	1 Mbps
	<NR1>	CAN bit rate in bps


```
Example      :BUS1:CAN:BITRate?
              >1000000

              :BUS1:CAN:BITRate rate800k
              :BUS1:CAN:BITRate?
              >800000

              :BUS1:CAN:BITRate 25000
              :BUS1:CAN:BITRate?
              >25000
```

Set →

→ Query

:BUS1:LIN:BITRate

Description Sets or returns the bit rate of the LIN bus.

Syntax :BUS1:LIN:BITRate {<NR1> | ?}

Parameter/Return parameter <NR1> LIN bit rate in bps.

```
Example      :BUS1:LIN:BITRate 9600
              Sets the LIN bit rate to 9600bps.
```

Set →

→ Query

:BUS1:LIN:IDFormat

Description Sets or returns the LIN ID format.

Syntax :BUS1:LIN:IDFormat {NOPARity|PARity|?}

Parameter/Return parameter NOPARity Don't include parity bits with Id.
PARity Include parity bits with Id.

```
Example      :BUS1:LIN:IDFormat?
              NOPARITY
              Returns the ID format.
```

Set →

→ Query

:BUS1:LIN:POLARity

Description Sets or returns the LIN polarity.

Syntax :BUS1:LIN:POLARity {NORMAL|INVerted|?}

Parameter/Return parameter NORMAL Normal LIN polarity

parameter	INVERTed	Inverted LIN polarity
-----------	----------	-----------------------

Example :BUS1:LIN:POLARity?
 NORMAL
 Returns the LIN polarity.

:BUS1:LIN:SAMPLEpoint → Query

Description Returns the sample point.

Syntax :BUS1:LIN:SAMPLEpoint?

Return Parameter Returns the sample point of the LIN bus as a percentage.

Example :BUS1:LIN:SAMPLEpoint?
 50
 Returns the sample point as a percentage.

Set →

:BUS1:LIN:SOURce → Query

Description Sets or returns the LIN data source.

Syntax :BUS1:LIN:SOURce {CH1 | CH2 | CH3 | CH4 | ? }

Parameter/Return parameter	CH1 ~ CH4	Analog channel source
----------------------------	-----------	-----------------------

Example :BUS1:LIN:SOURCE?
 >CH1
 Returns the LIN source.

Set →

:BUS1:LIN:STANDard → Query

Description Sets or returns the LIN standard.

Syntax :BUS1:LIN:STANDard {V1X|V2X|BOTH|?}

Parameter/Return parameter	V1X	Lin standard version 1.x
	V2X	Lin standard version 2.x

	BOTH	Both standards
--	-------------	----------------

Example

:BUS1:LIN:STANDard?

>BOTH

Returns the LIN standard.

Mark Commands

:MARK.....	178
:MARK:CREATE.....	178
:MARK:DELEte.....	179

:MARK



Description	Move to next or previous event mark.	
Syntax	:MARK { NEXT PREvious }	
Related commands	:MARK:CREATE :MARK:DELEte	
Parameter	NEXT	Move to next mark
	PREvious	Move to previous mark
Example	:MARK NEXT Moves to the next event mark.	

:MARK:CREATE



Description	Creates a mark on the waveform at the current position or creates a mark for all the events for the current waveform.	
Syntax	:MARK:CREATE { CURRent ALL }	
Related commands	:MARK :MARK:DELEte	
Parameter	CURRent	Creates a mark at the current position
	ALL	Creates a mark for all the events.
Example	:MARK:CREATE CURRent Creates a mark at the current position.	

:MARK:DELEte

Set →

Description	Deletes the current mark or all the marks on a waveform.	
Syntax	:MARK:DELEte { CURRent ALL }	
Related commands	:MARK :MARK:CREATE	
Parameter	CURRent	Deletes the current mark
	ALL	Deletes all the marks
Example	:MARK:DELEte CURRent Deletes the current mark.	

Search Commands

:SEARCH:COPY	181
:SEARCH:STATE	182
:SEARCH:TOTAL	182
:SEARCH:TRIGger:TYPe	182
:SEARCH:TRIGger:SOURce.....	183
:SEARCH:TRIGger:EDGE:SLOP	183
:SEARCH:TRIGger:LEVel	184
:SEARCH:TRIGger:HLEVel	184
:SEARCH:TRIGger:LLEVel	185
:SEARCH:TRIGger:PULSEWidth:POLarity	185
:SEARCH:TRIGger:RUNT:POLarity.....	186
:SEARCH:TRIGger:RISEFall:SLOP	186
:SEARCH:TRIGger:PULSE:WHEn.....	187
:SEARCH:TRIGger:PULSE:TIME.....	187
:SEARCH:TRIGger:RUNT:WHEn	188
:SEARCH:TRIGger:RUNT:TIME	188
:SEARCH:TRIGger:RISEFall:WHEn	189
:SEARCH:TRIGger:RISEFall:TIME	189
:SEARCH:TRIGger:BUS:TYPe.....	190
:SEARCH:TRIGger:BUS:B1:I2C:CONDition	190
:SEARCH:TRIGger:BUS:B1:I2C:ADDRes:MODE	191
:SEARCH:TRIGger:BUS:B1:I2C:ADDRes:TYPe	191
:SEARCH:TRIGger:BUS:B1:I2C:ADDRes:VALue.....	192
:SEARCH:TRIGger:BUS:B1:I2C:ADDRes	
:DIRection	193
:SEARCH:TRIGger:BUS:B1:I2C:DATa:SIZE	193
:SEARCH:TRIGger:BUS:B1:I2C:DATa:VALue	194
:SEARCH:TRIGger:BUS:B1:UART:CONDition.....	195
:SEARCH:TRIGger:BUS:B1:UART:RX:DATa:SIZE.....	196
:SEARCH:TRIGger:BUS:B1:UART:RX:DATa:VALue..	196
:SEARCH:TRIGger:BUS:B1:UART:TX:DATa:SIZE	197
:SEARCH:TRIGger:BUS:B1:UART:TX:DATa:VALue..	197

:SEARCH:TRIGger:BUS:B1:SPI:CONDition198
 :SEARCH:TRIGger:BUS:B1:SPI:DATa:SIze199
 :SEARCH:TRIGger:BUS:B1:SPI:DATa:MISO:VALue .199
 :SEARCH:TRIGger:BUS:B1:SPI:DATa:MOSI:VALue .200
 :SEARCH:TRIGger:BUS:B1:CAN:CONDition201
 :SEARCH:TRIGger:BUS:B1:CAN:FRAMeType202
 SEARCH:TRIGger:BUS:B1:CAN:IDentifier:MODE ...202
 :SEARCH:TRIGger:BUS:B1:CAN:IDentifier:VALue ...203
 :SEARCH:TRIGger:BUS:B1:CAN:IDentifier
 :DIRection203
 :SEARCH:TRIGger:BUS:B1:CAN:DATa:QUALifier....204
 :SEARCH:TRIGger:BUS:B1:CAN:DATa:SIze205
 :SEARCH:TRIGger:BUS:B1:CAN:DATa:VALue205
 :SEARCH:TRIGger:BUS:B1:LIN:CONDition206
 :SEARCH:TRIGger:BUS:B1:LIN:DATa:QUALifier207
 :SEARCH:TRIGger:BUS:B1:LIN:DATa:SIze208
 :SEARCH:TRIGger:BUS:B1:LIN:DATa:VALue208
 :SEARCH:TRIGger:BUS:B1:LIN:ERRTYPE209
 :SEARCH:TRIGger:BUS:B1:LIN:IDentifier:VALue.....209
 :SEARCH:FFTPeak:METhod210
 :SEARCH:FFTPeak:METhod:MPEak.....211
 :SEARCH:FFTPeak:SINFo.....211
 :SEARCH:FFTPeak:LIST212

:SEARCH:COPY



Description	Copies the search settings to the trigger settings or copies the trigger settings to the search settings.	
Syntax	:SEARCH:COPY {SEARCHtotrigger TRIGgertosearch}	
Parameter	SEARCHtotrigger	Copy the search setting to the trigger settings.
	TRIGgertosearch	Copy the trigger settings to the search settings.

Example :SEARCH:COPY SEARCHtotrigger
Copies the search settings to the trigger settings.

Set →

:SEARCH:STATE

→ Query

Description Sets or queries whether the Search function is on or off.

Syntax :SEARCH:STATE { OFF | ON | ? }

Parameter/Return parameter	OFF	Turn the Search function on.
	ON	Turn the Search function off.

Example :SEARCH:STATE ON
Turn Search on.

:SEARCH:TOTAL

→ Query

Description Returns the total number of events found from the search function.

Syntax :SEARCH:TOTAL?

Return parameter	<NR1>	Number of events.
------------------	-------	-------------------

Example :SEARCH:TOTAL?
5

Set →

:SEARCH:TRIGger:TYPE

→ Query


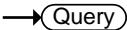
Description Sets or queries the search trigger type.

Syntax :SEARCH:TRIGger:TYPE { EDGe | PULSEWidth | RUNT | RISEFall | FFTPeak | LOGic | BUS | ? }

Parameter/Return parameter	EDGe	Edge trigger
	PULSEWidth	Pulse width trigger
	RUNT	Runt trigger
	RISEFall	Rise and Fall trigger

	FFTPeak	FFT Peak trigger
	LOGic	Logic trigger
	BUS	Bus trigger

Example :SEARCH:TRIGger:TYPE EDGE
Sets the search trigger to the edge type.

:SEARCH:TRIGger:SOURce  

Description Sets or queries the search trigger source.

Syntax :SEARCH:TRIGger:SOURce {CH1 | CH2 | CH3 | CH4|? }

Parameter/Return parameter	CH1 to CH4	Channel 1 to Channel 4
----------------------------	------------	------------------------

Example :SEARCH:TRIGger:SOURce CH1
Sets the search trigger source as CH1.

:SEARCH:TRIGger:EDGE:SLOP  

Description Sets or queries the search trigger slope.

Syntax :SEARCH:TRIGger:EDGE:SLOP { RISE | FALL | EITHER | ? }

Related commands :SEARCH:TRIGger:TYPE

Parameter	RISe	Rising slope
	FALL	Falling slope
	EITHer	Either rising or falling slope

Return parameter Returns the trigger slope.

Example :SEARCH:TRIGger:EDGE:SLOP FALL
Sets the search trigger slope to falling.

Set →

→ Query

:SEARCH:TRIGger:LEVel

Description	Sets or queries the search trigger level.	
Syntax	:SEARCH:TRIGger:LEVel {TTL ECL SETTO50 <NRf> ?}	
Related commands	:SEARCH:TRIGger:TYPe	
Parameter	<NRf>	Trigger level value
	TTL	Sets the search trigger level to TTL.
	ECL	Sets the search trigger level to ECL.
	SETTO50	Sets the search trigger level to the User level (50% by default).
Return parameter	<NR3>	Returns the trigger.
Example1	:SEARCH:TRIGger:LEVel TTL Sets the search trigger level to TTL.	
Example2	:SEARCH:TRIGger:LEVel 3.30E-1 Sets the search trigger level to 330mV/mA.	

Set →

→ Query


:SEARCH:TRIGger:HLEVel

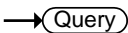
Description	Sets or queries the high level search trigger.	
Note	Applicable for Rise and Fall/Pulse Runt search triggers.	
Syntax	:SEARCH:TRIGger:HLEVel { <NRf> ?}	
Related commands	:SEARCH:TRIGger:TYPe	
Parameter	<NRf>	High level value.
Return parameter	<NR3>	Returns the high level search trigger.
Example	:SEARCH:TRIGger:HLEVel 3.30E-1 Sets the high level search trigger to 330mV/mA.	

:SEARCH:TRIGger:LLEVel




Description	Sets or queries the low level search trigger.	
Note	Applicable for Rise and Fall/Pulse Runt triggers.	
Syntax	:SEARCH:TRIGger:LLEVel { <NRf> ?}	
Related commands	:SEARCH:TRIGger:TYPe	
Parameter	<NRf>	Low level value.
Return parameter	<NR3>	Returns the low level.
Example	:SEARCH:TRIGger:LLEVel -3.30E-3 Sets the low level search trigger to 330mV/mA.	

:SEARCH:TRIGger:PULSEWidth:POLarity




Description	Sets or queries the pulse width search trigger polarity.	
Syntax	:SEARCH:TRIGger:PULSEWidth:POLarity {POSitive NEGative ?}	
Related commands	:SEARCH:TRIGger:TYPe	
Parameter	POSitive	Positive polarity
	NEGative	Negative polarity
Return parameter	Returns the pulse width polarity.	
Example	:SEARCH:TRIGger:PULSEWidth:POLarity POSitive Sets the pulse width polarity to positive.	

:SEARCH:TRIGger:RUNT:POLarity (Set) →
→ (Query)

Description	Sets or queries the Pulse Runt search trigger polarity.	
Syntax	:SEARCH:TRIGger:RUNT:POLarity {POSitive NEGative EITHer ?}	
Related commands	:SEARCH:TRIGger:TYPE	
Parameter	POSitive	Positive polarity
	NEGative	Negative polarity
	EITHer	Positive or negative polarity
Return parameter	Returns the pulse runt search trigger polarity.	
Example	:SEARCH:TRIGger:RUNT:POLarity POSitive Sets the Pulse Runt search trigger polarity to positive.	

:SEARCH:TRIGger:RISEFall:SLOP (Set) →
→ (Query)

Description	Sets or queries the slope of the Rise and Fall search trigger.	
Syntax	:SEARCH:TRIGger:RISEFall:SLOP { RISE FALL EITHer ? }	
Related commands	:SEARCH:TRIGger:TYPE	
Parameter	RISe	Rising slope
	FALL	Falling slope
	EITHer	Either rising or falling slope
Return parameter	Returns the rise & fall slope.	
Example	:SEARCH:TRIGger:RISEFall :SLOP RISE Sets the Rise & Fall search trigger slope to rising.	

Set →
→ Query

:SEARCH:TRIGger:PULSe:WHEn

Description	Sets or queries the pulse width search trigger conditions.								
Syntax	:SEARCH:TRIGger:PULSe:WHEn {MOREthan LESSthan EQual UNEQual ?}								
Related commands	:SEARCH:TRIGger:TYPE :SEARCH:TRIGger:PULSe:TIME								
Parameter	<table border="1"> <tr><td>MOREthan</td><td>></td></tr> <tr><td>LESSthan</td><td><</td></tr> <tr><td>EQual</td><td>=</td></tr> <tr><td>UNEQual</td><td>≠</td></tr> </table>	MOREthan	>	LESSthan	<	EQual	=	UNEQual	≠
MOREthan	>								
LESSthan	<								
EQual	=								
UNEQual	≠								
Return parameter	Returns the pulse width search trigger conditions.								
Example	:SEARCH:TRIGger:PULSe:WHEn UNEQual Sets the pulse width search trigger conditions to not equal to (≠).								

Set →
→ Query

:SEARCH:TRIGger:PULSe:TIME

Description	Sets or queries the pulse width search trigger time.
Syntax	:SEARCH:TRIGger:PULSe:TIME {<NRf> ?}
Related commands	:SEARCH:TRIGger:TYPE :SEARCH:TRIGger:PULSe:WHEn
Parameter	<NRf> Pulse width time (4ns~10s)
Return parameter	<NR3> Returns the pulse width time in seconds.
Example	:SEARCH:TRIGger:PULSe:TIME 4.00E-5 Sets the pulse width search trigger to 40.0us.

Set →

→ Query

:SEARCH:TRIGger:RUNT:WHEn

Description	Sets or queries the pulse runt search trigger conditions.
Syntax	:SEARCH:TRIGger:RUNT:WHEn {MOREthan LESSthan EQUAL UNEQUAL ? }
Related commands	:SEARCH:TRIGger:TYPE :SEARCH:TRIGger:RUNT:TIME
Parameter	MOREthan > LESSthan < Equal = UNEQUAL ≠
Return parameter	Returns the pulse runt search trigger conditions.
Example	:SEARCH:TRIGger:RUNT:WHEn UNEQUAL Sets the pulse runt search trigger condition to unequal (≠).

Set →

→ Query

:SEARCH:TRIGger:RUNT:TIME

Description	Sets or queries the pulse runt search trigger time.
Syntax	:SEARCH:TRIGger:RUNT:TIME {<NRf> ? }
Related commands	:SEARCH:TRIGger:TYPE :SEARCH:TRIGger:RUNT:WHEn
Parameter	<NRf> Pulse runt time (4nS to 10S)
Return Parameter	<NR3> Returns the runt time in seconds.
Example	:SEARCH:TRIGger:RUNT:TIME 4.00E-5 Sets the pulse runt time to 40.0uS.

Set →
 → Query

:SEARCH:TRIGger:RISEFall:WHEn

Description	Sets or queries the rise and fall search trigger conditions.												
Syntax	:SEARCH:TRIGger:RISEFall:WHEn {MOREthan LESSthan EQual UNEQual ? }												
Related commands	:SEARCH:TRIGger:TYPE :SEARCH:TRIGger:RISEFall:TIME												
Parameter	<table style="width: 100%; border-collapse: collapse;"> <tr><td style="width: 60%;">MOREthan</td><td style="width: 10%; text-align: center;">></td><td style="width: 30%;"></td></tr> <tr><td>LESSthan</td><td style="text-align: center;"><</td><td></td></tr> <tr><td>Equal</td><td style="text-align: center;">=</td><td></td></tr> <tr><td>UNEQual</td><td style="text-align: center;">≠</td><td></td></tr> </table>	MOREthan	>		LESSthan	<		Equal	=		UNEQual	≠	
MOREthan	>												
LESSthan	<												
Equal	=												
UNEQual	≠												
Return parameter	Returns the rise and fall search trigger condition.												
Example	:SEARCH:TRIGger:RISEFall:WHEn UNEQual Sets the rise and fall search trigger condition to unequal (≠).												

Set →
 → Query

:SEARCH:TRIGger:RISEFall:TIME

Description	Sets or queries the rise and fall time.
Syntax	:SEARCH:TRIGger:RISEFall:TIME {<NRf> ? }
Related commands	:SEARCH:TRIGger:TYPE :SEARCH:TRIGger:RISEFall:WHEn
Parameter	<NRf> Rise and Fall time (4nS to 10S)
Return Parameter	<NR3> Returns the rise and fall time in seconds.
Example	:SEARCH:TRIGger:RISEFall:TIME 4.00E-5 Sets the trigger rise and fall time to 40.0us.

:SEARCH:TRIGger:BUS:TYPE

→ Query

Description	Returns the current bus type.	
Syntax	:SEARCH:TRIGger:BUS:TYPE?	
Return parameter	I2C	I2C mode
	SPI	SPI mode
	UART	UART mode
	CAN	CAN mode
	LIN	LIN mode
	PARAllel	Parallel mode
Example	:SEARCH:TRIGger:BUS:TYPE? UART	

:SEARCH:TRIGger:BUS:B1:I2C:CONDition

Set →

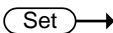
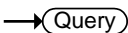
→ Query

Description	Sets or queries the I ² C search trigger conditions.	
Syntax	:SEARCH:TRIGger:BUS:B1:I2C:CONDition {START STOP REPEATstart ACKMISS ADDRess DATA ADDRANDDATA ? }	
Parameter	START	Set Start as the I ² C search trigger condition.
	STOP	Set Stop as the I ² C search trigger condition.
	REPEATstart	Set Repeat of Start as the I ² C search trigger condition.
	ACKMISS	Set Missing Acknowledgement as the I ² C search trigger condition.
	ADDRess	Set Address as the I ² C search trigger condition.

DATA	Set Data as the I ² C search trigger condition.
ADDRANDDATA	Set Address and Data as the I ² C search trigger condition.

Return parameter Returns the I²C bus search trigger condition.

Example :SEARCH:TRIGger:BUS:B1:I2C:CONDition ADDRess
Set Address as the I²C search trigger condition.

:SEARCH:TRIGger:BUS:B1:I2C:ADDRess 
:MODE 

Description Sets or queries the I²C addressing mode (7 or 10 bits) for the search trigger.


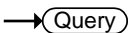
Syntax :SEARCH:TRIGger:BUS:B1:I2C:ADDRess:MODE
{ADDR7 | ADDR10 | ? }

Related commands :SEARCH:TRIGger:BUS:B1:I2C:CONDition

Parameter	ADDR7	7 bit addressing
	ADDR10	10 bit addressing

Return Parameter	0	7 bit addressing
	1	10 bit addressing

Example :SEARCH:TRIGger:BUS:B1:I2C:ADDRess:MODE?
0
The addressing mode is current set to 7 bits.

:SEARCH:TRIGger:BUS:B1:I2C:ADDRess 
:TYPE 

Description Sets the I²C bus address type, or queries what the setting is for the search trigger.

Syntax :SEARCH:TRIGger:BUS:B1:I2C:ADDRess:TYPE
{GENeralcall | STARtbyte | HSmode | EEPROM | CBUS
| ? }

Related commands :SEARCH:TRIGger:BUS:B1:I2C:CONDition

Parameter	GENeralcall	Set a general call address (0000 000 0).
	STARtbyte	Set a start byte address. (0000 000 1)
	HSmode	Set a high-speed mode address. (0000 1xx x)
	EEPROM	Set an EEPROM address. (1010 xxx x)
	CBUS	Set a CBUS address. (0000 001 x)

Return Parameter Returns the address type

Example :SEARCH:TRIGger:BUS:B1:I2C:ADDRes:TYPe? CBUS

:SEARCH:TRIGger:BUS:B1:I2C:ADDRes :VALue

Description Sets or queries the I²C bus address value when the I²C search trigger is set to trigger on Address or Address/Data.

Syntax :SEARCH:TRIGger:BUS:B1:I2C:ADDRes:VALue {<string> | ? }

Related commands :SEARCH:TRIGger:BUS:B1:I2C:ADDRes:MODE

Parameter	<string>	7/10 characters, must be enclosed in double quotes "string". x = don't care 1 = binary 1 0 = binary 0
-----------	----------	--

Return Parameter Returns the address value in binary.

Example 1 :SEARCH:TRIGger:BUS:B1:I2C:ADDRess:MODE
 ADDR7
 :SEARCH:TRIGger:BUS:B1:I2C:ADDRess:VALue
 "xxx0101"
 Sets the address to XXX0101

Example 2 :SEARCH:TRIGger:BUS:B1:I2C:ADDRess:VALue?
 XXX0101

:SEARCH:TRIGger:BUS:B1:I2C:ADDRess (Set) →
 :DIRection → (Query)

Description Sets or queries the address bit as read write or
 don't care for the search function.

Note This setting only applies when the I²C search
 trigger is set to trigger on Address or
 Address/Data

Syntax :SEARCH:TRIGger:BUS:B1:I2C:ADDRess:DIRection
 { READ | WRITE | NOCARE | ? }

Related commands :SEARCH:TRIGger:BUS:B1:I2C:CONDition

Parameter	READ	Set read as the data direction.
	WRITE	Set write as the data direction.
	NOCARE	Set either as the data direction.

Return Parameter Returns the direction (READ, WRITE, NOCARE).

Example :SEARCH:TRIGger:BUS:B1:I2C:ADDRess:DIRection
 READ
 Sets the direction to READ.

:SEARCH:TRIGger:BUS:B1:I2C:DATA:SIZE (Set) →
 → (Query)

Description Sets or queries the data size in bytes for the I²C
 bus.


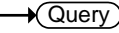
Note	This setting only applies when the I ² C search trigger is set to trigger on Data or Address/Data	
Syntax	:SEARCH:TRIGger:BUS:B1:I2C:DATA:SIZE {<NR1> ? }	
Related commands	:SEARCH:TRIGger:BUS:B1:I2C:CONDition	
Parameter	<NR1>	Number of data bytes (1 to 5).
Return parameter	<NR1>	Returns the number of bytes.
Example	:SEARCH:TRIGger:BUS:B1:I2C:DATA:SIZE 3 Sets the number of bytes to 3.	

(Set) →

:SEARCH:TRIGger:BUS:B1:I2C:DATA:VALue → (Query)

Description	Sets or queries the triggering data value for the I ² C bus when the I ² C search trigger is set to trigger on Data or Address/Data.	
Syntax	:SEARCH:TRIGger:BUS:B1:I2C:DATA:VALue {<string> ? }	
Related commands	:SEARCH:TRIGger:BUS:B1:I2C:DATA:SIZE	
Parameter	<string>	The number of characters in the string depends on the data size setting. The string must be enclosed in double quotes, "string". x = don't care 1 = binary 1 0 = binary 0
Return Parameter	Returns the data value.	
Example 1	:SEARCH:TRIGger:BUS:B1:I2C:DATA:SIZE 1 :SEARCH:TRIGger:BUS:B1:I2C:DATA:VALue "1x1x0101" Sets the value to XXX0101	

Example 2 :SEARCH:TRIGger:BUS:B1:I2C:DATA:VALue?
 1X1X0101

:SEARCH:TRIGger:BUS:B1:UART 
 :CONDition 


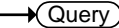
Description Sets or queries the UART search triggering condition.

Syntax :SEARCH:TRIGger:BUS:B1:UART:CONDition
 { RXSTArt | RXDATA | RXENDPacket | TXSTArt |
 TXDATA | TXENDPacket | TXPARItYerr | RXPARItYerr
 | ? }

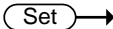
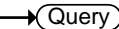
Parameter	RXSTArt	Set search trigger on the RX Start Bit.
	RXDATA	Set search trigger on RX Data.
	RXENDPacket	Set search trigger on the RX End of Packet condition.
	RXPARItYerr	Set search trigger on RX Parity error condition.
	TXSTArt	Set search trigger on the TX Start Bit.
	TXDATA	Set search trigger on TX Data.
	TXENDPacket	Set search trigger on the TX End of Packet condition.
	TXPARItYerr	Set search trigger on TX Parity error condition.

Return Parameter Returns the search triggering condition.

Example :SEARCH:TRIGger:BUS:B1:UART:CONDition TXDATA
 Sets the UART bus to trigger on Tx Data for the search function.

:SEARCH:TRIGger:BUS:B1:UART:RX:DATA:  →
 SIZE 

Description	Sets or queries the number of bytes for UART data.	
Note	This setting only applies when the UART search trigger is set to trigger on Rx Data	
Syntax	:SEARCH:TRIGger:BUS:B1:UART:RX:DATA:SIZE {<NR1> ?}	
Related commands	:SEARCH:TRIGger:BUS:B1:UART:CONDition	
Parameter	<NR1>	Number of bytes (1 to 10).
Return parameter	<NR1>	Returns the number of bytes.
Example	:SEARCH:TRIGger:BUS:B1:UART:RX:DATA:SIZE 5 Sets the number of bytes to 5.	


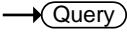
:SEARCH:TRIGger:BUS:B1:UART:RX:DATA:  →
 VALue 

Description	Sets or queries the search triggering data value for the UART bus when the bus is set to trigger on Rx Data.	
Syntax	:SEARCH:TRIGger:BUS:B1:UART:RX:DATA:VALue {<string> ?}	
Related commands	:SEARCH:TRIGger:BUS:B1:UART:RX:DATA:SIZE	
Parameter	<string>	The number of characters in the string depends on the data size setting. The string must be enclosed in double quotes, "string". x = don't care 1 = binary 1 0 = binary 0

Return Parameter Returns the data value.

Example1 :SEARCH:TRIGger:BUS:B1:UART:CONDition RXDATA
 :SEARCH:TRIGger:BUS:B1:UART:RX:DATA:SIZE 1
 :SEARCH:TRIGger:BUS:B1:UART:RX:DATA:VALue
 "1x1x0101"
 Sets the value to 1x1x0101

Example 2 :SEARCH:TRIGger:BUS:B1:UART:RX:DATA:VALue?
 1X1X0101

:SEARCH:TRIGger:BUS:B1:UART:TX:DATA 
 :SIze 

Description Sets or queries the number of bytes for UART data.

Note This setting only applies when the UART search trigger is set to trigger on Tx Data


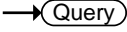
Syntax :SEARCH:TRIGger:BUS:B1:UART:TX:DATA:SIZE
 {<NR1> | ?}

Related commands :SEARCH:TRIGger:BUS:B1:UART:CONDition

Parameter <NR1> Number of bytes (1 to 10).

Return parameter <NR1> Returns the number of bytes.

Example :SEARCH:TRIGger:BUS:B1:UART:TX:DATA:SIZE 5
 Sets the number of bytes to 5.

:SEARCH:TRIGger:BUS:B1:UART:TX:DATA: VALue 


Description Sets or queries the search triggering data value for the UART bus when the bus is set to trigger on Tx Data.

Syntax :SEARCH:TRIGger:BUS:B1:UART:TX:DATA:VALue
 {<string> | ?}

Related commands	:SEARCH:TRIGger:BUS:B1:UART:TX:DATA:SIZE	
Parameter	<string>	<p>The number of characters in the string depends on the data size setting. The string must be enclosed in double quotes, "string".</p> <p>x = don't care</p> <p>1 = binary 1</p> <p>0 = binary 0</p>

Return Parameter Returns the data value.

Example 1 :SEARCH:TRIGger:BUS:B1:UART:CONDition TXDATA
 :SEARCH:TRIGger:BUS:B1:UART:TX:DATA:SIZE 1
 :SEARCH:TRIGger:BUS:B1:UART:TX:DATA:VALue
 "1x1x0101"
 Sets the value to 1x1x0101

Example 2 :SEARCH:TRIGger:BUS:B1:UART:TX:DATA:VALue?
 1X1X0101

Set →
 → Query

Description	Sets or queries the SPI search triggering condition.	
Syntax	:SEARCH:TRIGger:BUS:B1:SPI:CONDition {SS MISO MOSI MISOMOSI ? }	
Parameter	SS	Set to trigger on the Slave Selection condition.
	MISO	Set to trigger on the Master-In Slave-Out condition.
	MOSI	Set to trigger on the Master-Out Slave-In condition.
	MISOMOSI	Set to trigger on the Master-In Slave-Out and Master-Out Slave-In conditions.

Return Parameter Returns the triggering condition.

Example :SEARCH:TRIGger:BUS:B1:SPI:CONDition MISO
Sets the SPI bus to trigger on MISO.

:SEARCH:TRIGger:BUS:B1:SPI:DATA:SIZE  

Description Sets or queries the number of words for SPI data for the search function.

Note This setting only applies when the SPI search trigger is set to trigger on MISO, MOSI or MISO/MOSI

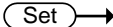
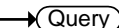
Syntax :SEARCH:TRIGger:BUS:B1:SPI:DATA:SIZE {<NR1> | ?}

Related commands :SEARCH:TRIGger:BUS:B1:SPI:CONDition

Parameter <NR1> Number of words (1 to 32).

Return parameter <NR1> Returns the number of words.

Example :SEARCH:TRIGger:BUS:B1:SPI:DATA:SIZE 10
Sets the number of words to 10.

:SEARCH:TRIGger:BUS:B1:SPI:DATA:MISO:  

Description Sets or queries the search triggering data value for the SPI bus when the bus is set to trigger on MISO or MISO/MOSI.

Syntax :SEARCH:TRIGger:BUS:B1:SPI:DATA:MISO:VALue {<string> | ?}

Related commands :SEARCH:TRIGger:BUS:B1:SPI:DATA:SIZE


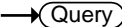
Parameter <sting> The number of characters in the string depends on the data size setting. The string must be enclosed in double quotes, "string".

x = don't care
 1 = binary 1
 0 = binary 0

Return Parameter Returns the data value.

Example 1 :SEARCH:TRIGger:BUS:B1:SPI:CONDition MISO
 :SEARCH:TRIGger:BUS:B1:SPI:DATA:SIZE 2
 :SEARCH:TRIGger:BUS:B1:SPI:DATA:MISO:VALue
 "1x1x0101"
 Sets the value to 1x1x0101

Example 2 :SEARCH:TRIGger:BUS:B1:SPI:DATA:MISO:VALue?
 1X1X0101

:SEARCH:TRIGger:BUS:B1:SPI:DATA:MOSI: 
 VALue 

Description Sets or queries the search triggering data value for the SPI bus when the bus is set to trigger on MOSI or MISO/MOSI.

Syntax :SEARCH:TRIGger:BUS:B1:SPI:DATA:MOSI:VALue
 {<string> | ? }

Related commands :SEARCH:TRIGger:BUS:B1:SPI:DATA:SIZE

Parameter <sting> The number of characters in the string depends on the data size setting. The string must be enclosed in double quotes, "string".
 x = don't care
 1 = binary 1
 0 = binary 0

Return Parameter Returns the data value.

Example1 :SEARCH:TRIGger:BUS:B1:SPI:CONDition MOSI
 :SEARCH:TRIGger:BUS:B1:SPI:DATA:SIZE 2
 :SEARCH:TRIGger:BUS:B1:SPI:DATA:MOSI:VALue
 "1x1x0101"
 Sets the value to 1x1x0101

Example2 :SEARCH:TRIGger:BUS:B1:SPI:DATA:MOSI:VALue?
 1X1X0101

Set →

:SEARCH:TRIGger:BUS:B1:CAN:CONDition → Query

Description Sets or returns the CAN search trigger condition.

Syntax :SEARCH:TRIGger:BUS:B1:CAN:CONDition
 {SOF|FRAMeType|Identifier|DATA|IDANDDATA|EOF|
 ACKMISS|STUFFERR|?}

Parameter/ Return parameter	SOF	Sets search to trigger on a start of frame
	FRAMeType	Sets search to trigger on the type of frame
	Identifier	Sets search to trigger on a matching identifier
	DATA	Sets search to trigger on matching data
	IDANDDATA	Sets search to trigger on matching identifier and data field
	EOF	Sets search to trigger on the end of frame
	ACKMISS	Sets search to trigger on a missing acknowledge
	STUFFERR	Sets search to trigger on a bit stuffing error

Example1 :SEARCH:TRIGger:BUS:B1:CAN:CONDition SOF
 Triggers search on a start of frame.

Example2 :SEARCH:TRIGger:BUS:B1:CAN:CONDition?
>SOF

Set →

:SEARCH:TRIGger:BUS:B1:CAN:FRAMEType → Query

Description Sets or returns the frame type for the CAN FRAMEType search trigger.

Syntax :SEARCH:TRIGger:BUS:B1:CAN:FRAMEType {DATA|REMOte|ERRor|OVERLoad|?}

Parameter/ Return parameter	DATA	Sets the frame type to data frame
	REMOte	Sets the frame type to remote frame
	ERRor	Sets the frame type to error frame
	OVERLoad	Sets the frame type to overload

Example :SEARCH:TRIGger:BUS:B1:CAN:FRAMEType DATA
Sets the frame type to DATA.

SEARCH:TRIGger:BUS:B1:CAN:IDentifier: MODE Set →
→ Query

Description Sets or returns the CAN identifier mode for the bus.

Syntax :SEARCH:TRIGger:BUS:B1:CAN:IDentifier:MODE {STANDard|EXTended|?}

Parameter/ Return parameter	STANDard	Standard addressing mode
	EXTended	Extended addressing mode

Example :SEARCH:TRIGger:BUS:B1:CAN:IDentifier:MODE?
>STANDARD

:SEARCH:TRIGger:BUS:B1:CAN:IDentifier:MODE
EXTENDED

:SEARCH:TRIGger:BUS:B1:CAN:IDentifier:MODE?
>EXTENDED

:SEARCH:TRIGger:BUS:B1:CAN:IDentifier: VALue (Set) →
→ (Query)

Description Sets or returns the identifier string used for the CAN search trigger.

Note: Only applicable when the search trigger condition is set to ID or IDANDDATA.

Syntax :SEARCH:TRIGger:BUS:B1:CAN:IDentifier:VALue {<string>|?}

Related Commands :SEARCH:TRIGger:BUS:B1:CAN:IDentifier:MODE

Parameter/ Return parameter	<code><string></code>	The size of the string depends on the data size setting. The string must be enclosed in double quotes, "string". String contents: x = don't care 1 = binary 1 0 = binary 0
--	-----------------------------	--

Example :SEARCH:TRIGger:BUS:B1:CAN:CONDition ID
:SEARCH:TRIGger:BUS:B1:CAN:IDentifier:MODE STANDARD
:SEARCH:TRIGger:BUS:B1:CAN:IDentifier:VALue "01100X1X01X"
:SEARCH:TRIGger:BUS:B1:CAN:IDentifier:VALue? >01100X1X01X

:SEARCH:TRIGger:BUS:B1:CAN:IDentifier: DIRection (Set) →
→ (Query)

Description Sets or queries the address bit as read, write or don't care.

Syntax :SEARCH:TRIGger:BUS:B1:CAN:IDentifier:DIRection {READ|WRITE|NOCARE?}

Parameter/ Return parameter	READ	Sets read as the data direction
	WRITE	Sets write as the data direction
	NO CARE	Sets either as the data direction

```
Example2 :SEARCH:TRIGger:BUS:B1:CAN:IDentifier:DIRection?
>WRITE

:SEARCH:TRIGger:BUS:B1:CAN:IDentifier:DIRection
READ

:SEARCH:TRIGger:BUS:B1:CAN:IDentifier:DIRection?
>READ
```

```
:SEARCH:TRIGger:BUS:B1:CAN:DATA:
QUALifier
```

Description Sets or returns the CAN data qualifier.
 Note: Only applicable when the search triggering condition is set to DATA or IDANDDATA.

Syntax :SEARCH:TRIGger:BUS:B1:CAN:DATA:QUALifier
 {LESSthan|MOREthan|EQUAL|UNEQUAL|LESSEQUAL|M
 OREEQUAL|?}

Parameter/ Return parameter	LESSthan	Sets search to trigger when the data is less than the qualifier value.
	MOREthan	Sets search to trigger when the data is greater than the qualifier value.
	EQUAL	Sets search to trigger when the data is equal to the qualifier value.
	UNEQUAL	Sets search to trigger when the data is not equal to the qualifier value.
	LESSEQUAL	Sets search to trigger when the data is less than or equal to the qualifier value.
	MOREEQUAL	Sets search to trigger when the data is more than or equal to the qualifier value.

Example :SEARCH:TRIGger:BUS:B1:CAN:DATA:QUALifier?
>EQUAL
:SEARCH:TRIGger:BUS:B1:CAN:DATA:QUALifier
MOREthan
:SEARCH:TRIGger:BUS:B1:CAN:DATA:QUALifier?
>MORETHAN

Set →

:SEARCH:TRIGger:BUS:B1:CAN:DATA:SIZE → Query

Description Sets or returns the length of the data string in bytes for the CAN search trigger.

Note: Only applicable when the condition is set to DATA or IDANDDATA.

Syntax :SEARCH:TRIGger:BUS:B1:CAN:DATA:SIZE {<NR1>|?}

Parameter/
Return parameter <NR1> 1~8 (bytes)

Example :SEARCH:TRIGger:BUS:B1:CAN:DATA:SIZE?
>1
:SEARCH:TRIGger:BUS:B1:CAN:DATA:SIZE 2
:SEARCH:TRIGger:BUS:B1:CAN:DATA:SIZE?
>2

:SEARCH:TRIGger:BUS:B1:CAN:DATA:
VALue Set →
→ Query

Description Sets or returns the binary data string to be used for the CAN search trigger.

Related
Commands :SEARCH:TRIGger:BUS:B1:CAN:DATA:SIZE

Syntax :SEARCH:TRIGger:BUS:B1:CAN:DATA:VALue
{<string>|?}

Parameter/
Return parameter <string> The size of the string depends on the data size setting. The string must be enclosed in double quotes, "string".

String contents:
 x = don't care
 1 = binary 1
 0 = binary 0

Example :SEARCH:TRIGger:BUS:B1:CAN:DATA:SIZE 1
 :SEARCH:TRIGger:BUS:B1:CAN:DATA:VALue
 "01010X1X"
 :SEARCH:TRIGger:BUS:B1:CAN:DATA:VALue?
 >01010X1X

Set →

:SEARCH:TRIGger:BUS:B1:LIN:CONDition → Query

Description Sets or returns the LIN search trigger condition.

Syntax :SEARCH:TRIGger:BUS:B1:LIN:CONDition
 {SYNCFieLd|IDentifier|DATA|IDANDDATA|WAKEup|SLEEPEE|ERRor|?}

Parameter/ Return parameter	SYNCFieLd	Sets the LIN search trigger condition to the sync field.
	IDentifier	Sets the LIN search trigger condition to identifier field.
	DATA	Sets the LIN search trigger condition to the data field.
	IDANDDATA	Sets the LIN search trigger condition to identifier and data field
	WAKEup	Sets the LIN search trigger condition to wake up.
	SLEEPEE	Sets the LIN search trigger condition to sleep.
	ERRor	Sets the LIN search trigger condition to error.


```
Example      :SEARCH:TRIGger:BUS:B1:LIN:CONDition?
              >IDANDDATA
              :SEARCH:TRIGger:BUS:B1:LIN:CONDition DATA
              :SEARCH:TRIGger:BUS:B1:LIN:CONDition?
              >DATA
```


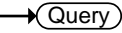
```
:SEARCH:TRIGger:BUS:B1:LIN:DATA:          (Set) →
QUALifier                                  → (Query)
```

Description Sets or returns the LIN data qualifier.
 Note: Only applicable when the search trigger condition is set to DATA or IDANDDATA.

Syntax :SEARCH:TRIGger:BUS:B1:LIN:DATA:QUALifier
 {LESSthan|MOREthan|EQual|UNEQual|LESSEQual|M
 OREEQual|?}

Parameter/ Return parameter	LESSthan	Sets search to trigger when the data is less than the qualifier value.
	MOREthan	Sets search to trigger when the data is greater than the qualifier value.
	EQual	Sets search to trigger when the data is equal to the qualifier value.
	UNEQual	Sets search to trigger when the data is not equal to the qualifier value.
	LESSEQual	Sets search to trigger when the data is less than or equal to the qualifier value.
	MOREEQual	Sets search to trigger when the data is more than or equal to the qualifier value.

Example :SEARCH:TRIGger:BUS:B1:LIN:DATA:QUALifier?
>EQUAL
:SEARCH:TRIGger:BUS:B1:LIN:DATA:QUALifier
MOREthan
:SEARCH:TRIGger:BUS:B1:LIN:DATA:QUALifier?
>MORETHAN

:SEARCH:TRIGger:BUS:B1:LIN:DATA:SIZE  


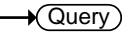
Description Sets or returns the length of the data string in bytes for the LIN search trigger.

Note: Only applicable when the condition is set to DATA or IDANDDATA.

Syntax :SEARCH:TRIGger:BUS:B1:LIN:DATA:SIZE {<NR1>|?}

Parameter/ Return parameter	<NR1>	1~8 (bytes)
--------------------------------	-------	-------------

Example :SEARCH:TRIGger:BUS:B1:LIN:DATA:SIZE?
>1
:SEARCH:TRIGger:BUS:B1:LIN:DATA:SIZE 2
:SEARCH:TRIGger:BUS:B1:LIN:DATA:SIZE?
>2

:SEARCH:TRIGger:BUS:B1:LIN:DATA:
VALue  

Description Sets or returns the binary data string to be used for the LIN search trigger.

Note: Only applicable when the condition is set to DATA or IDANDDATA.

Related Commands :SEARCH:TRIGger:BUS:B1:LIN:DATA:SIZE

Syntax :SEARCH:TRIGger:BUS:B1:LIN:DATA:VALue
{<string>|?}

Parameter/ Return parameter	<string>	The size of the string depends on the data size setting. The string must be enclosed in double quotes, "string". String contents: x = don't care 1 = binary 1 0 = binary 0
--------------------------------	----------	--

Example :SEARCH:TRIGger:BUS:B1:LIN:DATA:SIZE 1
 :SEARCH:TRIGger:BUS:B1:LIN:DATA:VALue
 "01010X1X"
 :SEARCH:TRIGger:BUS:B1:LIN:DATA:VALue?
 >01010X1X

Set →
 → Query

Description Sets or returns the error type to be used for the LIN search trigger.

Syntax :SEARCH:TRIGger:BUS:B1:LIN:ERRTYPE
 {SYNC|PARItY|CHecksum|?}

Parameter/ Return parameter	SYNC	Sets the LIN error type to SYNC.
	PARItY	Sets the LIN error type to parity.
	CHecksum	Sets the LIN error type to checksum.

Example :SEARCH:TRIGger:BUS:B1:LIN:ERRTYPE?
 >SYNC
 :SEARCH:TRIGger:BUS:B1:LIN:ERRTYPE CHECKSUM
 :SEARCH:TRIGger:BUS:B1:LIN:ERRTYPE?
 >CHECKSUM

Set →
 → Query

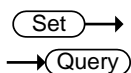
Description	Sets or returns the identifier string to be used for the LIN search trigger. Note: Only applicable when the condition is set to ID or IDANDDATA.	
Syntax	:SEARCH:TRIGger:BUS:B1:LIN:IDentifier:VALue {<string> ?}	
Parameter/ Return parameter	<string>	The size of the string depends on the data size setting. The string must be enclosed in double quotes, "string". String contents: x = don't care 1 = binary 1 0 = binary 0
Example	:SEARCH:TRIGger:BUS:B1:LIN:CONDition ID :SEARCH:TRIGger:BUS:B1:LIN:IDentifier:VALue "00X1X01X" :SEARCH:TRIGger:BUS:B1:LIN:IDentifier:VALue? >01100X1X01X	

:SEARCH:FFTPeak:METHod (Set) →
→ (Query)

Description	Sets or returns the FFT peak method type.	
Related Commands	:SEARCH:TRIGger:TYPE :SEARCH:FFTPeak:METHod:MPEak :SEARCH:TRIGger:LEVel	
Syntax	:SEARCH:FFTPeak:METHod {MPEak LEVel ?}	
Parameter/ Return parameter	MPEak	Sets the peak method to the Max Peak type.
	LEVel	Sets the peak methods to the Level type.

```
Example      :SEARCH:FFTPeak:MEthod LEVEL
              :SEARCH:FFTPeak:MEthod?
              >LEVEL
              :SEARCH:TRIGger:LEVel?
              >1.000E+00
              :SEARCH:TRIGger:LEVel 2
              :SEARCH:TRIGger:LEVel?
              >2.000E+00
```

:SEARCH:FFTPeak:MEthod:MPEak



Description Sets the active peak number (1 ~ 10) or return the frequency of the active peak number.

Related Commands :SEARCH:TRIGger:TYPe
:SEARCH:FFTPeak:MEthod

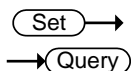
Syntax :SEARCH:FFTPeak:MEthod:MPEak {<NR1> | ?}

Parameter <NR1> Active peak number.

Return parameter <NR3> Frequency of the active peak.

```
Example      :SEARCH:FFTPeak:MEthod MPEak
              :SEARCH:FFTPeak:MEthod?
              >MPEAK
              :SEARCH:FFTPeak:MEthod:MPEak?
              >1.000E+00
              :SEARCH:FFTPeak:MEthod:MPEak 2
              :SEARCH:FFTPeak:MEthod:MPEak?
              >2.000E+00
```

:SEARCH:FFTPeak:SINFo



Description Sets or returns the State Info to “Mark” or “Peak”.

Related Commands	:SEARCH:TRIGger:TYPe	
Syntax	:SEARCH:FFTPeak:SINFo {MARK PEAK ?}	
Parameter/ Return parameter	MARK	Sets the State Info to Mark.
	PEAK	Sets the State Info to Peak.
Example	<pre>:SEARCH:FFTPeak:SINFo? >PEAK :SEARCH:FFTPeak:SINFo mark :SEARCH:FFTPeak:SINFo? >MARK</pre>	

:SEARCH:FFTPeak:LIST



Description	Returns the data of the search event table.	
Syntax	:SEARCH:FFTPeak:LIST?	
Example	<pre>:SEARCH:FFTPeak:LIST? No.,Frequency,Value; 1,1.000E+04,-6.400E+00; 2,2.750E+06,-7.360E+01; 3,2.830E+06,-7.280E+01; 4,2.910E+06,-7.200E+01; 5,3.020E+06,-7.120E+01; 6,3.170E+06,-7.040E+01; 7,5.550E+06,-8.240E+01; 8,5.640E+06,-8.160E+01; 9,5.740E+06,-8.080E+01; 10,5.900E+06,-8.000E+01;</pre>	

Label Commands

:CHANnel<X>:LABel.....	213
:CHANnel<X>:LABel:DISPlay	214
:REF<X>:LABel	214
:REF<X>:LABel:DISPlay	215
:BUS1:LABel.....	216
:BUS1:LABel:DISPlay	216
:SET<X>:LABel	217

:CHANnel<X>:LABel




Description	Sets or returns the file label for the selected channel.	
Syntax	:CHANnel<X>:LABel {<string> ?}	
Related commands	:CHANnel<X>:LABel:DISPlay	
Parameter	<X>	Channel 1, 2, 3, 4
	<string>	The string must be no more than 8 characters and only contain alphanumeric characters in addition to period, dash and underscore characters. The string must be enclosed in double quotes, "string".
Return parameter	<string>	Returns the label for the selected channel. No return indicates that there has not been a file label assigned for the selected channel.
Example1	:CHANnel1:LABel "CH1_lab" Sets the channel 1 label as "CH1_lab".	
Example2	:CHANnel1:LABel? CH1_lab	

Set →

→ Query

:CHANnel<X>:LABel:DISPlay

Description	Turns the label on/off for the selected channel or returns its status.	
Syntax	:CHANnel<X>:LABel:DISPlay { OFF ON ? }	
Related commands	:CHANnel<X>:LABel	
Parameter	<X>	Channel 1, 2, 3, 4
	OFF	Turns the file label off for the selected channel.
	ON	Turns the file label on for the selected channel.
Return parameter	Returns the status of the file label for the selected channel (ON, OFF).	

Example

```
:CHANnel1:LABel "CH1"
:CHANnel1:LABel:DISPlay ON
:CHANnel1:LABel:DISPlay?
ON
```

Sets the channel 1 label to "CH1" and then turns the label display on. The query return shows that the label is on.

Set →

→ Query

:REF<X>:LABel

Description	Sets or returns the file label for the selected reference waveform.	
Syntax	:REF<X>:LABel {<string> ?}	
Related commands	:REF<X>:LABel:DISPlay	
Parameter	<X>	REF 1, 2, 3, 4

	<string>	The string must be no more than 8 characters and only contain alphanumeric characters in addition to period, dash and underscore characters. The string must be enclosed in double quotes, "string".
Return parameter	<string>	Returns the label for the selected reference waveform. No return indicates that there has not been a file label assigned for the selected reference waveform.

Example1 :REF1:LABel "REF1_lab"
 Sets the REF1 label as "REF1_lab".

Example2 :REF1:LABel?
 REF1_lab

:REF<X>:LABel:DISPlay 

Description Turns the label on/off for the selected reference waveform or returns its status.

Syntax :REF<X>:LABel:DISPlay { OFF | ON | ? }

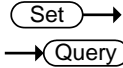
Related commands :REF<X>:LABel

Parameter	<X>	Reference waveform 1, 2, 3, 4
	OFF	Turns the file label off for the selected reference waveform.
	ON	Turns the file label on for the selected reference waveform.

Return parameter Returns the status of the file label for the selected reference waveform (ON, OFF).

Example :REF1:LABel "REF1"
 :REF1:LABel:DISPlay ON
 :REF1:LABel:DISPlay?
 ON

Sets the label for reference waveform 1 to "REF1" and then turns the label display on. The query return shows that the label is on.

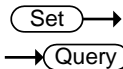


:BUS1:LABel

Description	Sets or returns the file label for the bus.	
Syntax	:BUS1:LABel {<string> ?}	
Related commands	:BUS1:LABel:DISPlay	
Parameter	<string>	The string must be no more than 8 characters and only contain alphanumeric characters in addition to period, dash and underscore characters. The string must be enclosed in double quotes, "string".
Return parameter	<string>	Returns the label for the bus. No return indicates that there has not been a file label assigned for bus.

Example1 :BUS1:LABel "Bus"
 Sets the bus label as "Bus".

Example2 :BUS1:LABel?
 Bus



:BUS1:LABel:DISPlay

Description	Turns the label on/off for the bus or returns its status.	
Syntax	:BUS1:LABel:DISPlay { OFF ON ? }	

Related commands	:BUS1:LABel	
Parameter	OFF	Turns the file label off for the bus.
	ON	Turns the file label on for the bus.
Return parameter	Returns the status of the file label for the bus (ON, OFF).	
Example	:BUS1:LABel "Bus"	
	:BUS1:LABel:DISPlay ON :BUS1:LABel:DISPlay? ON	
	Sets the label for the bus to "Bus" and then turns the label display on. The query return shows that the label is on.	





Description	Sets or returns the file label for the selected setup.	
Syntax	:SET<X>:LABel {<string> ?}	
Related commands	:SET<X>:LABel:DISPlay	
Parameter	<X>	Setup number 1 to 20
	<string>	The string must be no more than 8 characters and only contain alphanumeric characters in addition to period, dash and underscore characters. The string must be enclosed in double quotes, "string".
Return parameter	<string>	Returns the label for the selected setup. No return indicates that there has not been a file label assigned for the selected setup.
Example1	:SET1:LABel "SET1_lab"	
	Sets the label for setup 1 as "SET1_lab".	

Example2 :SET1:LABel?
 SET1_lab

Segment Commands

:SEGMENTS:STATE	219
:SEGMENTS:CURRENT	220
:SEGMENTS:TOTALNUM	220
:SEGMENTS:TIME	221
:SEGMENTS:DISPALL	221
:SEGMENTS:MEASURE:MODE	221
:SEGMENTS:MEASURE:PLOT:SOURCE	222
:SEGMENTS:MEASURE:PLOT:DIVIDE	222
:SEGMENTS:MEASURE:PLOT:SELECT	223
:SEGMENTS:MEASURE:PLOT:RESULTS.....	223
:SEGMENTS:MEASURE:TABLE:SOURCE	224
:SEGMENTS:MEASURE:TABLE:SELECT.....	224
:SEGMENTS:MEASURE:TABLE:LIST.....	225
:SEGMENTS:MEASURE:TABLE:SAVE	225
:SEGMENTS:SAVE	225
:SEGMENTS:SAVE:SOURCE	226
:SEGMENTS:SAVE:SELECT:START	226
:SEGMENTS:SAVE:SELECT:END	227

		
		
:SEGMENTS:STATE		
Description	Turns the segmented memory function on/off or queries its state.	
Syntax	:SEGMENTS:STATE { OFF ON ? }	
Related commands	:RUN :STOP	
Parameter/ Return parameter	OFF	Turns the segmented memory off.
	ON	Turns the segmented memory on.
Example1	:SEGMENTS:STATE ON Turns segmented memory on.	

Set →

→ Query

:SEGMents:CURRent

Description Sets or queries the current segment. The total number of segments depends on the record length.

Syntax :SEGMents:CURRent
{SETTOMIN|SETTOMAX|<NR1>|?}

Related commands :SEGMents:STATE
:SEGMents:TOTAlnum

Parameter/ Return parameter	SETTOMIN	Current segment = min segment
	SETTOMAX	Current segment = max segment
	<NR1>	1~29000

Example1 :SEGMents:CURRent 10
Sets the current segment to segment number 10.

Set →

→ Query

:SEGMents:TOTAlnum

Description Sets or queries the total number of segments for the segmented memory function. The total number of segments depends on the record length.

Syntax :SEGMents:TOTAlnum
{SETTOMIN|SETTOMAX|<NR1>|?}

Related commands :SEGMents:STATE
:SEGMents:CURRent

Parameter/ Return parameter	SETTOMIN	Sets to the minimum number
	SETTOMAX	Sets to the maximum number
	<NR1>	1~29000

Example1 :SEGMents:TOTAlnum SETTOMAX
Sets the number of segments to max number (29000).

:SEGMents:TIME

→ Query

Description	Returns the time of the current segment in relation to the first segment.
Syntax	:SEGMents:TIME?
Related commands	:SEGMents:STATE :SEGMents:CURRent
Return parameter	The segment time as <NR3>.
Example	:SEGMents:TIME? >8.040E-03 Returns the segment time.

:SEGMents:DISPALL

Set →

→ Query

Description	Sets or queries whether all the segments are displayed on the screen.
Syntax	:SEGMents:DISPALL {OFF ON ?}
Related commands	:SEGMents:STATE :SEGMents:CURRent
Parameter/ Return parameter	OFF Turns the display all function off. ON Turns the display all function on.
Example1	:SEGMents:DISPALL ON Turns the display all function on.

:SEGMents:MEASure:MODE

Set →

→ Query

Description	Sets or queries the measurement mode.
Syntax	:SEGMents:MEASure:MODE {OFF PLOT TABLE ?}
Related commands	:MEASurement:MEAS<x>

Parameter/ Return parameter	OFF	Disables the automatic measurement function for the segments measurement.
	PLOT	Sets the measurement mode to Statistics.
	TABLE	Sets the measurement mode to a measurement list.

Example :SEGMents:MEASure:MODE?>PLOT
Returns the measurement mode as Statistics.

:SEGMents:MEASure:PLOT:SOURce Set →
→ Query

Description	Sets or queries the statistics source.	
Syntax	:SEGMents:MEASure:PLOT:SOURce {<NR1> ? }	
Related commands	:SEGMents:MEASure:MODE :SEGMents:MEASure:PLOT:DIVide :SEGMents:MEASure:PLOT:SElect :SEGMents:MEASure:PLOT:RESults	

Parameter/ Return parameter	<NR1>	1~8 (Automatic measurement item 1~8)
--------------------------------	-------	--------------------------------------

Example1 :SEGMents:MEASure:PLOT:SOURce 1
Sets the source as auto measurement item 1.

:SEGMents:MEASure:PLOT:DIVide Set →
→ Query

Description	Sets or queries the number of bins for the statistics function.	
Syntax	:SEGMents:MEASure:PLOT:DIVide {<NR1> ? }	
Related commands	:SEGMents:MEASure:PLOT:SOURce :SEGMents:MEASure:PLOT:SElect	

Parameter/ Return parameter	<NR1>	1~20
--------------------------------	-------	------

Example1 :SEGMents:MEASure:PLOT:DIVide 5
Sets the number of bins to 5 for the statistics function.

:SEGMents:MEASure:PLOT:SElect (Set) →
← (Query)

Description Sets or queries which bin to view the statics of.

Syntax :SEGMents:MEASure:PLOT:SElect {<NR1> | ? }

Related commands :SEGMents:MEASure:PLOT:SOURce
:SEGMents:MEASure:PLOT:DIVide

Parameter	<NR1>	1~20 (cannot exceed the number of bins)
-----------	-------	--

Return parameter Return the bin number as <NR3>.

Example1 :SEGMents:MEASure:PLOT:SElect 5
Set to bin number 5.

:SEGMents:MEASure:PLOT:RESults ← (Query)

Description Returns the results of the currently selected bin for the statistics measurement.

Note At least one automatic measurement must be turned on.

Syntax :SEGMents:MEASure:PLOT:RESults?

Related commands :SEGMents:STATE
:SEGMents:MEASure:MODE PLOT
:SEGMents:MEASure:PLOT:SOURce
:SEGMents:MEASure:PLOT:DIVide
:SEGMents:MEASure:PLOT:SElect

Return parameter Returns the statistics measurements as a string.

Example :SEGMents:STATE ON
 STOP
 :SEGMents:MEASure:MODE PLOT
 :SEGMents:MEASure:PLOT:SOURce 1
 :SEGMents:MEASure:PLOT:DIVide 10
 :SEGMents:MEASure:PLOT:SElect 1
 :SEGMents:MEASure:PLOT:RESults?
 > MAX,1.000kHz;MIN,1.000kHz;MEAN,1.000kHz;
 Bin Statistics,1 of 10;Percent,10.00%;Count,1;
 Measured,10;Unmeasured,0;Bin Range,
 1.000kHz~1.000kHz;
 Plots the results for automatic measurement #1,
 bin 1 of 10.

:SEGMents:MEASure:TABLE:SOURce Set →
→ Query

Description Sets or queries the source of the measurement list.

Syntax :SEGMents:MEASure:TABLE:SOURce {CH1 | CH2 | CH3 | CH4 | ? }

Related commands :SEGMents:MEASure:MODE
 :SEGMents:MEASure:TABLE:SElect
 :SEGMents:MEASure:TABLE:LIST

Parameter/ Return parameter CH1~CH4 Channel 1 to 4

Example1 :SEGMents:MEASure:TABLE:SOURce CH1
 Sets the source to CH1.

:SEGMents:MEASure:TABLE:SElect Set →
→ Query

Description Sets or queries a segment to view in the measurement table.

Syntax :SEGMents:MEASure:TABLE:SElect {<NR1> | ? }

Related commands :SEGMents:TOTAlnum

Parameter <NR1> 1~29000

Return parameter Returns the number of segments as <NR3>.

Example1 :SEGMents:MEASure:TABLE:SElect 10
Select segment number 10.

:SEGMents:MEASure:TABLE:LIST → Query

Description Returns the measurement results of each segment in the list.

Syntax :SEGMents:MEASure:TABLE:LIST?

Return parameter Returns the measurements results as a string for each segment.

Example :SEGMents:MEASure:TABLE:LIST?
>"GW MDO-2074E, serial number PXXXXXX,
version V1.11",Segment Summary : CH1, Seg.,Pk-
Pk (V),Pk-Pk (V),1,8.00m,8.00m.....etc

:SEGMents:MEASure:TABLE:SAVE Set →

Description Saves the list of segment automatic measurement results.

Syntax :SEGMents:MEASure:TABLE:SAVE

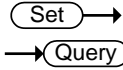
:SEGMents:SAVE Set →

Description Saves the segments.

Syntax :SEGMents:SAVE

Related Commands :SEGMents:SAVE:SOURce
:SEGMents:SAVE:SElect:START
:SEGMents:SAVE:SElect:END

Example :SEGMents:SAVe:SOURce CH1
 :SEGMents:SAVe:SELEct:STARt 1
 :SEGMents:SAVe:SELEct:END 10
 :SEGMents:SAVe



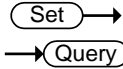
:SEGMents:SAVe:SOURce

Description Sets or queries the source segment waveform to save.

Syntax :SEGMents:SAVe:SOURce {CH1 | CH2 | CH3 | CH4 |? }

Parameter/ Return parameter CH1~CH4 Channel 1 to 4.

Example :SEGMents:SAVe:SOURce CH1
 >Sets the source to CH1.



:SEGMents:SAVe:SELEct:STARt


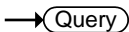
Description Sets or queries the starting segment to save from. The number of possible segments depends on the record length.

Syntax :SEGMents:SAVe:SELEct:STARt {SETTOMIN | SETTOMAX | <NR1> | ? }

Related commands :SEGMents:TOTalnum

Parameter/ Return parameter SETTOMIN Sets the starting segment to min segment.
 SETTOMAX Sets the starting segment to the max segment.
 <NR1> Sets the segment to 1~29000



Example :SEGMents:SAVe:SELEct:STARt 2
 Sets the starting segment to segment number 2.



		 
:SEGMents:SAVe:SElect:END		
Description	Sets or queries the ending segment to save from. The number of possible segments depends on the record length.	
Syntax	:SEGMents:SAVe:SElect:END {SETTOMIN SETTOMAX <NR1> ? }	
Related commands	:SEGMents:TOTalnum	
Parameter/ Return parameter	SETTOMIN	Sets the starting segment to min segment.
	SETTOMAX	Sets the starting segment to the max segment.
	<NR1>	Sets the segment to 1~29000.
Return parameter	<NR3>	Returns the ending segment as NR3.
Example	:SEGMents:SAVe:SElect:END 10 Sets the ending segment to segment number 10.	

DVM Commands

The DVM commands are only available when the optional DVM software is installed.

:DVM:STATE	228
:DVM:SOURce	228
:DVM:MODE	229
:DVM:VALue	229

					
:DVM:STATE					
Description	Sets or queries the DVM state to on or off.				
Syntax	:DVM:STATE {OFF ON ? }				
Related commands	:DVM:SOURce :DVM:MODE				
Parameter/ Return parameter	<table border="1"> <tr> <td>OFF</td> <td>Turns the DVM off.</td> </tr> <tr> <td>ON</td> <td>Turns the DVM on.</td> </tr> </table>	OFF	Turns the DVM off.	ON	Turns the DVM on.
OFF	Turns the DVM off.				
ON	Turns the DVM on.				
Example	:DVM:STATE ON Turns the DVM state on.				

			
:DVM:SOURce			
Description	Sets or queries the source of the DVM.		
Syntax	:DVM:SOURce {CH1 CH2 CH3 CH4 ?}		
Related commands	:DVM:STATE :DVM:MODE		
Parameter/ Return parameter	<table border="1"> <tr> <td>CH1~CH4</td> <td>Channel 1 to 4.</td> </tr> </table>	CH1~CH4	Channel 1 to 4.
CH1~CH4	Channel 1 to 4.		
Example	:DVM:SOURce CH1 Sets the DVM source to channel 1.		

Set →
 → Query

:DVM:MODE

Description	Sets or queries the DVM mode.	
Syntax	:DVM:MODE {ACRMS DC DCRMS DUTY FREQUENCY ?}	
Related commands	:DVM:SOURce :DVM:STATE	
Parameter/ Return parameter	ACRMS	Sets the mode to AC RMS
	DC	Sets the mode to DC
	DCRMS	Sets the mode to DC RMS
	DUTY	Sets the mode to AC Duty
	FREQUENCY	Sets the mode to AC frequency
Example	:DVM:MODE DUTY Sets the DVM mode to DUTY.	

→ Query

:DVM:VALue

Description	Returns the measurement value of the selected mode.	
Syntax	:DVM:VALue?	
Related commands	:DVM:SOURce :DVM:STATE :DVD:MODE	
Return parameter	Returns the measurement value as <NR3>.	
Example	:DVM:VALue? >8.410E-04 Returns the measurement.	

Go_NoGo Commands

The GoNoGo APP must first be launched (or use the command, “:GONogo:SCRipt”) before any of the Go_NoGo or Template commands can be used.

:GONogo:CLear.....	230
:GONogo:EXECute	230
:GONogo:FUNCTion.....	231
:GONogo:NGCount	231
:GONogo:NGDefine.....	231
:GONogo:SOURce	232
:GONogo:VIOLation	232
:GONogo:SCRipt	232
:TEMPlate:MODE	233
:TEMPlate:MAXimum.....	233
:TEMPlate:MINimum.....	233
:TEMPlate:POSition:MAXimum.....	234
:TEMPlate:POSition:MINimum	234
:TEMPlate:SAVe:MAXimum	234
:TEMPlate:SAVe:MINimum	235
:TEMPlate:TOLerance.....	235
:TEMPlate:SAVe:AUTo	235

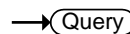
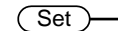
:GONogo:CLear



Description Clears the Go/NoGo counter.

Syntax :GONogo:CLear

:GONogo:EXECute



Description Enables or disables the Go/NoGo function or queries its state.

Syntax :GONogo:EXECute {OFF|ON|?}

Parameter/	OFF	Disabled
Return Parameter	ON	Enabled

Example :GONogo:EXECute OFF
Turns Go/NoGo off.

:GONogo:FUNcTion (Set) →

Description Initializes the Go/NoGo APP. This must be run after the Go/NoGo APP has been started.

Syntax :GONogo:FUNcTion

:GONogo:NGCount → (Query)

Description Returns the Go/NoGo counter.

Syntax :GONogo:NGCount{?}

Return parameter Returns a string in the following format “number of violations,total tests”

Example :GONogo:NGCount?
> 3,25
Indicates that 3 violations occurred over 25 tests.


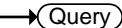
:GONogo:NGDefine (Set) →
→ (Query)

Description Sets the Go/NoGo “When” conditions.


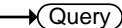
Syntax :GONogo:NGDefine {EXITs|ENTers?}

Parameter/	EXITs	Sets the NoGo condition to when the input signal exceeds the limit boundary.
Return Parameter	ENTers	Sets the NoGo condition to when the input signal stays within the limit boundary.


Example :GONogo:NGDefine EXITs
Sets the Go/NoGo condition to EXITs.

 →
 → 


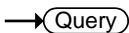
:GONogo:SOURce	
Description	Sets the source for the Go/NoGo signal.
Syntax	:GONogo:SOURce {CH1 CH2 CH3 CH4 ?}
Parameter/ Return Parameter	CH1~CH4
Example	:GONogo:SOURce CH1 Sets the source to CH1.


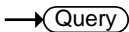
 →
 → 


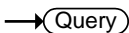
:GONogo:VIOLation					
Description	Sets or returns actions for the Go/NoGo violations.				
Syntax	:GONogo:VIOLation {STOP CONTInue ?}				
Parameter/ Return Parameter	<table border="0"> <tr> <td>STOP</td> <td>The waveform will be frozen.</td> </tr> <tr> <td>CONTInue</td> <td>Ignore the violation.</td> </tr> </table>	STOP	The waveform will be frozen.	CONTInue	Ignore the violation.
STOP	The waveform will be frozen.				
CONTInue	Ignore the violation.				
Example	:GONogo:VIOLation STOP Sets violation action to STOP.				

 →

:GONogo:SCRipt					
Description	Activates/Deactivates the Go/NoGo APP or queries its state.				
Syntax	:GONogo:SCRipt {OFF ON ?}				
Parameter/ Return Parameter	<table border="0"> <tr> <td>ON</td> <td>Turns Go/NoGo APP on.</td> </tr> <tr> <td>OFF</td> <td>Turns the Go/NoGo APP off.</td> </tr> </table>	ON	Turns Go/NoGo APP on.	OFF	Turns the Go/NoGo APP off.
ON	Turns Go/NoGo APP on.				
OFF	Turns the Go/NoGo APP off.				
Example	:GONogo:SCRipt? >ON The Go/NoGo script is on.				

	 						
:TEMPlate:MODE							
Description	Sets or returns the Go/NoGo template mode.						
Syntax	:TEMPlate:MODE {MAXimum MINimum AUTO ?}						
Parameter/ Return Parameter	<table style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 20%; background-color: #e0e0e0;">MAXimum</td> <td>Maximum template</td> </tr> <tr> <td style="background-color: #e0e0e0;">MINimum</td> <td>Minimum template</td> </tr> <tr> <td style="background-color: #e0e0e0;">AUTO</td> <td>Auto template</td> </tr> </table>	MAXimum	Maximum template	MINimum	Minimum template	AUTO	Auto template
MAXimum	Maximum template						
MINimum	Minimum template						
AUTO	Auto template						
Example	:TEMPlate:MODE AUTO Sets the template mode to AUTO.						

	 				
:TEMPlate:MAXimum					
Description	Defines or queries which waveform memory (REF1 or W1~W20) is set to the maximum template.				
Syntax	:TEMPlate:MAXimum {REF1 W1~W20 ?}				
Parameter/ Return Parameter	<table style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 20%; background-color: #e0e0e0;">REF1</td> <td>Reference one</td> </tr> <tr> <td style="background-color: #e0e0e0;">W1~W20</td> <td>Waveform memory 1 to 20</td> </tr> </table>	REF1	Reference one	W1~W20	Waveform memory 1 to 20
REF1	Reference one				
W1~W20	Waveform memory 1 to 20				
Example	:TEMPlate:MAXimum REF1 Saves the maximum template to REF1.				

	 				
:TEMPlate:MINimum					
Description	Defines or queries which waveform memory (REF1 or W1~W20) is set to the minimum template.				
Syntax	:TEMPlate:MINimum {REF2 W1~W20 ?}				
Parameter/ Return Parameter	<table style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 20%; background-color: #e0e0e0;">REF2</td> <td>Reference one</td> </tr> <tr> <td style="background-color: #e0e0e0;">W1~W20</td> <td>Waveform memory 1 to 20</td> </tr> </table>	REF2	Reference one	W1~W20	Waveform memory 1 to 20
REF2	Reference one				
W1~W20	Waveform memory 1 to 20				
Example	:TEMPlate:MINimum REF2 Saves the minimum template to REF2.				

:TEMPlate:POSition:MAXimum (Set) →
→ (Query)

Description	Sets or queries the position of the maximum template.	
Syntax	:TEMPlate:POSition:MAXimum {<NR2> ?}	
Parameter	<NR2>	Desired template position (-12.0 ~ +12.0 divisions)
Return parameter	Returns the position in the following format: “<NR2>Div”	
Example	:TEMPlate:POSition:MAXimum 3.00 Sets the maximum template position to 3.00 divisions.	

:TEMPlate:POSition:MINimum (Set) →
→ (Query)

Description	Sets or queries the position of the minimum template.	
Syntax	:TEMPlate:POSition:MINimum {<NR2> ?}	
Parameter	<NR2>	Desired template position (-12.0 ~ +12.0 divisions)
Return parameter	Returns the position in the following format: “<NR2>Div”	
Example	:TEMPlate:POSition:MINimum 3.00 Sets the minimum template position to 3.00 divisions.	

:TEMPlate:SAVe:MAXimum (Set) →

Description	Saves the maximum template.	
Syntax	:TEMPlate:SAVe:MAXimum	

:TEMPlate:SAVe:MINimum (Set) →

Description Saves the maximum template.

Syntax :TEMPlate:SAVe:MINimum

(Set) →

:TEMPlate:TOLerance → (Query)

Description Sets or queries the tolerance as a percentage.

Syntax :TEMPlate:TOLerance {<NR2>|?}

Parameter/ Return Parameter	<NR2>	The auto tolerance range (0.4% ~ 40%)
--------------------------------	-------	---------------------------------------

Example :TEMPlate:TOLerance 10
Sets the tolerance to 10%.

:TEMPlate:SAVe:AUTO (Set) →

Description Saves the AUTO template (maximum and minimum templates).

Syntax :TEMPlate:SAVe:AUTO

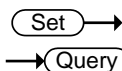
AWG Commands

The Arbitrary Wave Generator is available on the MDO-2000EX only.

:AWG<x>:AMPlitude.....	237
:AWG<x>:FREQuency.....	237
:AWG<x>:FUNctIon	238
:AWG<x>:OFFSet	239
:AWG<x>:OUTPut:LOAd:IMPEDance	239
:AWG<x>:OUTPut:STATE	239
:AWG<x>:PHAsE.....	240
:AWG<x>:PULSe:DUTYcycle	240
:AWG<x>:RAMP:SYMmetry	240
:AWG<x>:MODulation:STATE	241
:AWG<x>:MODulation:TYPe.....	241
:AWG<x>:MODulation:AM:DEPth	241
:AWG<x>:MODulation:AM:FREQ.....	242
:AWG<x>:MODulation:AM:SHApe.....	242
:AWG<x>:MODulation:AM:PHAsE	242
:AWG<x>:MODulation:AM:DUTYcycle	243
:AWG<x>:MODulation:AM:SYMmetry	243
:AWG<x>:MODulation:AM:RATE	244
:AWG<x>:MODulation:FM:DEV	244
:AWG<x>:MODulation:FM:FREQ	245
:AWG<x>:MODulation:FM:SHApe	245
:AWG<x>:MODulation:FM:PHAsE	245
:AWG<x>:MODulation:FM:DUTYcycle.....	246
:AWG<x>:MODulation:FM:SYMmetry	246
:AWG<x>:MODulation:FM:RATE.....	246
:AWG<x>:MODulation:FSK:FREQ	247
:AWG<x>:MODulation:FSK:RATE.....	247
:AWG<x>:SWEep:STATE	248
:AWG<x>:SWEep:TYPe	248
:AWG<x>:SWEep:START	248
:AWG<x>:SWEep:STOP.....	249

:AWG<x>:SWEep:TIME.....249
 :AWG<x>:SWEep:SPAN.....249
 :AWG<x>:SWEep:CENTer250
 :AWG<x>:ARBitrary:EDIT:NUMPOINT250
 :AWG<x>:ARBitrary:EDIT:FUNCTion.....250
 :AWG<x>:ARBitrary:SAVe:WAVEform251
 :AWG<x>:ARBitrary:LOAd:WAVEform251
 :AWG<x>:ARBitrary:EDIT:COPIY.....252
 :AWG<x>:ARBitrary:EDIT:CLEar.....252
 :AWG<x>:ARBitrary:EDIT:LINE253
 :AWG<x>:ARBitrary:EDIT:SCALE.....253
 :AWG<x>:ARBitrary:EDIT:POINT.....253
 :AWG<x>:ARBitrary:EDIT:POINT:ADD.....254
 :AWG<x>:ARBitrary:EDIT:POINT:DELEte.....254

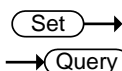
:AWG<x>:AMPlitude



Description	Sets or returns the waveform amplitude.	
Syntax	:AWG<x>:AMPlitude {<NRf> ?}	
Related command	:AWG<x>:OUTPut:LOAd:IMPEDance	
Parameter/ Return parameter	<x> <NRf>	Channel number 1~2. Amplitude in Volts. (50Ω impedance 0.1~2.5V) (High Z impedance 0.2~5V)

Example :AWG1:AMP 1

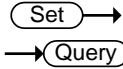
:AWG<x>:FREQuency



Description	Sets or returns the waveform frequency.	
Syntax	:AWG<x>:FREQuency {<NRf> ?}	

Parameter/	<x>	Channel number 1~2.
Return parameter	<NRf>	Frequency in Hertz.

Example :AWG1:FREQ 2000



:AWG<x>:FUNcTION

Description Sets or returns the type of waveform.

Syntax :AWG<x>:FUNcTION {ARBitrary | SINE | SQUAre | PULSe | RAMP | DC | NOISe | SINC | GAUSsian | LORENTz | EXPRise | EXPFall | HAVERSINe | CARDIac | ?}

Parameter/	<x>	Channel number 1~2.
Return parameter	ARBitrary	Arbitrary waveform
	SINE	Sine waveform
	SQUAre	Square waveform
	PULSe	Pulse waveform
	RAMP	Ramp waveform
	DC	DC waveform
	NOISe	Noise waveform
	SINC	Sinc waveform
	GAUSsian	Gaussian waveform
	LORENTz	Lorentz waveform
	EXPRise	Exponential rise waveform
	EXPFall	Exponential fall waveform
	HAVERSINe	Haversine waveform
	CARDIac	Cardiac waveform

Example :AWG1:FUNc?
>SINE

 →
 → 

:AWG<x>:OFFSet

Description	Sets or returns the waveform offset.	
Syntax	:AWG<x>:OFFSet {<NRf> ?}	
Parameter/ Return parameter	<x>	Channel number 1~2.
	<NRf>	Offset in Volts.
Example	:AWG1:OFFS	

 →
 → 

:AWG<x>:OUTPut:LOAD:IMPEDance

Description	Sets or returns the output termination	
Syntax	:AWG<x>:OUTPut:LOAD:IMPEDance {FIFTy HIGHZ ?}	
Parameter/ Return parameter	<x>	Channel number 1~2
	FIFTy	50 Ohm output termination
	HIGHZ	High Z output termination
Example	:AWG1:OUTP:LOA:IMPED HIGHZ Sets the output termination of channel 1 to high impedance.	

 →
 → 

:AWG<x>:OUTPut:STATE

Description	Sets or returns the channel output state.	
Syntax	:AWG<x>:OUTPut:STATE {OFF ON ?}	
Parameter/ Return parameter	<x>	Channel number 1~2
	OFF	Turns the channel output off
	ON	Turns the channel output on
Example	:AWG1:OUTP:STATE OFF Turns the channel 1 output off.	

Set →

:AWG<x>:PHase

→ Query

Description	Sets or returns the channel phase.	
Syntax	:AWG<x>:PHase {<NRf> ?}	
Parameter/ Return parameter	<x>	Channel number 1~2.
	<NRf>	Phase in degree -180~180°
Example	:AWG1:PHA 45 Sets the channel 1 phase to 45°.	

Set →

:AWG<x>:PULSe:DUTYcycle

→ Query

Description	Sets or returns the pulse duty cycle.	
Syntax	:AWG<x>:PULSe:DUTYcycle {<NRf> ?}	
Parameter/ Return parameter	<x>	Channel number 1~2.
	<NRf>	Duty cycle in percentage 0.2~99.8%
Example	:AWG1:PULS:DUTY 50 Sets the channel 1 pulse duty cycle to 50%.	

Set →

:AWG<x>:RAMP:SYMmetry

→ Query

Description	Sets or returns the ramp symmetry.	
Syntax	:AWG<x>:RAMP:SYMmetry {<NRf> ?}	
Parameter/ Return parameter	<x>	Channel number 1~2.
	<NRf>	Symmetry of the ramp waveform 0~100%
Example	:AWG1:RAMP:SYM 15 Sets the channel 1 ramp symmetry to 15%.	

Set →
 → Query

:AWG<x>:MODulation:STATE

Description	Sets or returns the modulation state.	
Syntax	:AWG<x>:MODulation:STATE {OFF ON ?}	
Parameter/ Return parameter	<x>	Channel number 1~2.
	OFF	Sets the modulation to off.
	ON	Sets the modulation to on.
Example	:AWG1:MOD:STATE ON Turns the modulation on for channel 1.	

Set →
 → Query

:AWG<x>:MODulation:TYPE

Description	Sets or returns the type of modulation.	
Syntax	:AWG<x>MODulation:TYPE {AM FM FSK ?}	
Parameter/ Return parameter	<x>	Channel number 1~2.
	AM	Sets a AM modulation.
	FM	Sets a FM modulation.
	FSK	Sets a FSK modulation.
Example	:AWG1:MOD:TYPE AM Sets a AM modulation for channel 1.	

Set →
 → Query

:AWG<x>:MODulation:AM:DEPth

Description	Sets or returns the AM modulation depth.	
Syntax	:AWG<x>:MODulation:AM:DEPth {<NRF> ?}	
Parameter/ Return parameter	<x>	Channel number 1~2.
	<NRF>	AM depth in percentage 0~120%.
Example	:AWG1:MOD:AM:DEP? >1.20000e+02	

:AWG<x>:MODulation:AM:FREQ (Set) →
→ (Query)

Description	Sets or returns the AM modulation frequency.	
Syntax	:AWG<x>:MODulation:AM:FREQ {<NRF> ?}	
Parameter/ Return parameter	<x>	Channel number 1~2.
	<NRF>	AM frequency in Hertz.
Example	:AWG1:MOD:AM:FREQ 1000 Sets the AM frequency to 1kHz.	

:AWG<x>:MODulation:AM:SHApe (Set) →
→ (Query)


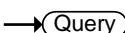
Description	Sets or returns the shape of the AM modulation.	
Syntax	:AWG<x>:MODulation:AM:SHApe {SINE SQUare PULSe RAMP NOISe ?}	
Parameter/ Return parameter	<x>	Channel number 1~2.
	SINE	Sine wave shape.
	SQUare	Square wave shape.
	PULSe	Pulse wave shape.
	RAMP	Ramp wave shape.
	NOISe	Noise wave shape.
Example	:AWG1:MOD:AM:SHA RAMP Sets a ramp shape to the AM modulating waveform.	

:AWG<x>:MODulation:AM:PHase (Set) →
→ (Query)

Description	Sets or returns the phase of the AM modulation (sine wave shape only).	
Syntax	:AWG<x>:MODulation:AM:PHase {<NRF> ?}	
Parameter/	<x>	Channel number 1~2.

Return parameter <NRf> Phase in degree -180~180°.

Example :AWG1:MOD:AM:PHA?
>-1.80000e+02

:AWG<x>:MODulation:AM:DUTYcycle  

Description Sets or returns the duty cycle of the AM modulation (pulse wave shape only).

Syntax :AWG<x>:MODulation:AM:DUTYcycle {<NRf> | ?}

Parameter/ Return parameter <x> Channel number 1~2.
<NRf> Duty cycle in percentage 2~98%.

Example :AWG1:MOD:AM:DUTY 50
Sets the duty cycle of the AM modulating waveform to 50%.

:AWG<x>:MODulation:AM:SYMmetry  

Description Sets or returns the symmetry of the AM modulation (ramp wave shape only).

Syntax :AWG<x>:MODulation:AM:SYMmetry {<NRf> | ?}

Parameter/ Return parameter <x> Channel number 1~2.
<NRf> Symmetry in percentage 0~100%.

Example :AWG1:MOD:AM:SYM 50
Sets the symmetry of the AM modulating waveform to 50%.

Set →

→ Query

:AWG<x>:MODulation:AM:RATE

Description	Sets or returns the rate of the AM modulation (noise wave shape only).	
Syntax	:AWG<x>:MODulation:AM:RATE {RATE10M RATE5M RATE1M RATE500K RATE100K RATE50K RATE10K RATE5K RATE1K ?}	
Parameter/ Return parameter	<x>	Channel number 1~2.
	RATE10M	10MHz noise rate.
	RATE5M	5MHz noise rate.
	RATE1M	1MHz noise rate.
	RATE500K	500kHz noise rate.
	RATE100K	100kHz noise rate.
	RATE50K	50kHz noise rate.
	RATE10K	10kHz noise rate.
	RATE5K	5kHz noise rate.
	RATE1K	1kHz noise rate.

Example :AWG1:MOD:AM:RATE RATE5K
Sets the noise rate of the AM modulating waveform to 5kHz.

Set →

→ Query

:AWG<x>:MODulation:FM:DEV

Description	Sets or returns the deviation of the FM modulation.	
Syntax	:AWG<x>:MODulation:FM:DEV {<NRf> ?}	
Parameter/ Return parameter	<x>	Channel number 1~2.
	<NRf>	Frequency deviation in Hertz.

Example :AWG1:MOD:FM:DEV?
>2.000000000e+02

:AWG<x>:MODulation:FM:FREQ
 →
 →

Description	Sets or returns the frequency of the FM modulation.	
Syntax	:AWG<x>:MODulation:FM:FREQ {<NRf> ?}	
Parameter/ Return parameter	<x>	Channel number 1~2.
	<NRf>	Frequency in Hertz.
Example	:AWG1:MOD:FM:FREQ 1000 Sets the frequency of the FM modulating waveform to 1kHz.	

:AWG<x>:MODulation:FM:SHApe
 →
 →

Description	Sets or returns the shape of the FM modulation.	
Syntax	:AWG<x>:MODulation:FM:SHApe {SINE SQUare PULSe RAMP NOISe ?}	
Parameter/ Return parameter	<x>	Channel number 1~2.
	SINE	Sine wave shape.
	SQUare	Square wave shape.
	PULSe	Pulse wave shape.
	RAMP	Ramp wave shape.
	NOISe	Noise wave shape.
Example	:AWG1:MOD:FM:SHA SINE Sets a sine shape to the FM modulation.	

:AWG<x>:MODulation:FM:PHAsE
 →
 →

Description	Sets or returns the phase of the FM modulation (sine wave shape only).	
Syntax	:AWG<x>:MODulation:FM:PHAsE {<NRf> ?}	
Parameter/	<x>	Channel number 1~2.

Return parameter <NRf> Phase in degree -180~180°.

Example :AWG1:MOD:FM:PHA 90
 Sets a 90° phase to the FM modulating waveform.

(Set) →
 → (Query)

:AWG<x>:MODulation:FM:DUTYcycle

Description Sets or returns the duty cycle of the FM modulation (pulse shape wave only).

Syntax :AWG<x>:MODulation:FM:DUTYcycle {<NRf> | ?}

Parameter/ Return parameter <x> Channel number 1~2.
 <NRf> Duty cycle in percentage 1~99%.

Example :AWG1:MOD:FM:DUTY 50
 Sets the duty cycle of the FM modulating waveform to 50%.

(Set) →
 → (Query)

:AWG<x>:MODulation:FM:SYMmetry

Description Sets or returns the symmetry of the FM modulation (ramp shape wave only).

Syntax :AWG<x>:MODulation:FM:SYMmetry {<NRf> | ?}

Parameter/ Return parameter <x> Channel number 1~2.
 <NRf> Symmetry in percentage 0~100%.

Example :AWG1:MOD:FM:SYM 50
 Sets the symmetry of the FM modulating waveform to 50%.

(Set) →
 → (Query)

:AWG<x>:MODulation:FM:RATE

Description Sets or returns the noise rate of the FM modulation (noise shape wave only).

Syntax :AWG<x>:MODulation:FM:RATE {RATE10M | RATE5M | RATE1M | RATE500K | RATE100K | RATE50K | RATE10K | RATE5K | RATE1K | ?}

Parameter/ Return parameter	<x>	Channel number 1~2.
	RATE10M	10MHz noise rate.
	RATE5M	5MHz noise rate.
	RATE1M	1MHz noise rate.
	RATE500K	500kHz noise rate.
	RATE100K	100kHz noise rate.
	RATE50K	50kHz noise rate.
	RATE10K	10kHz noise rate.
	RATE5K	5kHz noise rate.
	RATE1K	1kHz noise rate.

Example :AWG1:MOD:FM:RATE RATE5K
Sets the noise rate of the FM modulating waveform to 5kHz.

:AWG<x>:MODulation:FSK:FREQ Set →
→ Query

Description Sets or returns the hop frequency of the FSK modulation.

Syntax :AWG<x>:MODulation:FSK:FREQ {<Nrf> | ?}

Parameter/ Return parameter	<x>	Channel number 1~2.
	<Nrf>	Frequency in Hertz.

Example :AWG1:MOD:FSK:FREQ 2000000
Sets the FSK hop frequency to 2MHz.

:AWG<x>:MODulation:FSK:RATE Set →
→ Query

Description Sets or returns the FSK modulation rate.

Syntax :AWG<x>:MODulation:FSK:RATE {<Nrf> | ?}

Parameter/	<x>	Channel number 1~2.
Return parameter	<NRF>	Frequency in Hertz.

Example :AWG1:MOD:FSK:RATE 100000
Sets the FSK rate to 100kHz.

Set →

:AWG<x>:SWEep:STATE

→ Query

Description Sets or returns the Sweep mode state.

Syntax :AWG<x>:SWEep:STATE {OFF | ON | ?}

Parameter/	<x>	Channel number 1~2.
Return parameter	OFF	Sets the sweep mode to off.
	ON	Sets the sweep mode to on.

Example :AWG1:SWE:STATE ON
Turns the sweep mode to on for channel 1.

Set →

:AWG<x>:SWEep:TYPe

→ Query

Description Sets or returns the sweep mode type.

Syntax :AWG<x>:SWEep:TYPe {LINEAR | LOG | ?}

Parameter/	<x>	Channel number 1~2.
Return parameter	LINEAR	Sets the sweep mode to linear.
	LOG	Sets the sweep mode to logarithmic.

Example :AWG1:SWE:TYP LIN
Sets the sweep mode to linear for channel 1.

Set →

:AWG<x>:SWEep:START

→ Query

Description Sets or returns the start frequency of the sweep mode.

Syntax :AWG<x>:SWEep:START {<NRF> | ?}

Parameter/	<x>	Channel number 1~2.
------------	-----	---------------------

Return parameter `<NRf>` Start frequency in Hertz.

Example `:AWG1:SWE:START 1000`
 Sets the sweep mode start frequency to 1kHz.

`:AWG<x>:SWEep:STOP`
 →
 →

Description Sets or returns the stop frequency of the sweep mode.

Syntax `:AWG<x>:SWEep:STOP {<NRf> | ?}`

Parameter/
Return parameter `<x>` Channel number 1~2.
`<NRf>` Stop frequency in Hertz.

Example `:AWG1:SWE:STOP 500000`
 Sets the sweep mode stop frequency to 500kHz.

`:AWG<x>:SWEep:TIME`
 →
 →

Description Sets or returns the sweep time.

Syntax `:AWG<x>:SWEep:TIME {<NRf> | ?}`

Parameter/
Return parameter `<x>` Channel number 1~2.
`<NRf>` Sweep time in seconds.

Example `:AWG1:SWE:TIM 6.500e-01`
 Sets the sweep time to 650ms.

`:AWG<x>:SWEep:SPAN`
 →
 →

Description Alternatively to setting the start and stop frequencies, the span and center frequency can be set.

Syntax `:AWG<x>:SWEep:SPAN {<NRf> | ?}`

Parameter/
Return parameter `<x>` Channel number 1~2.
`<NRf>` Span of the sweep in Hertz.

Example :AWG1:SWE:SPAN 1100
Sets the span of the sweep to 1.1kHz.

:AWG<x>:SWEep:CENTer (Set) →
→ (Query)

Description Alternatively to setting the start and stop frequencies, the span and center frequency can be set.

Syntax :AWG<x>:SWEep:CENTer {<NRF> | ?}

Parameter/	<x>	Channel number 1~2.
Return parameter	<NRF>	Center frequency of the sweep in Hertz.

Example :AWG1:SWE:CENT 550
Sets the center frequency of the sweep to 550Hz.

:AWG<x>:ARBitrary:EDIT:NUMPOINT (Set) →
→ (Query)

Description Sets or returns the number of points of an arbitrary waveform.

Syntax :AWG<x>:ARBitrary:EDIT:NUMPOINT { <NR1> | ?}

Parameter/	<x>	Channel number 1~2.
Return parameter	<NR1>	Number of points

Example :AWG1:ARB:EDIT:NUMPOIN 1500
Sets 1500 points for the arbitrary waveform.

:AWG<x>:ARBitrary:EDIT:FUNCTion (Set) →

Description Sets the inbuilt waveform of the arbitrary waveform.

Syntax :AWG<x>:ARBitrary:EDIT:FUNCTion { SINE | SQUare | PULSe | RAMP | NOISe}

Parameter	<x>	Channel number 1~2.
	SINE	Sine wave shape.

SQUare	Square wave shape.
PULSe	Pulse wave shape.
RAMP	Ramp wave shape.
NOISe	Noise wave shape.

Example :AWG1:ARB:EDIT:FUNCT RAMP
Sets a ramp shape to the arbitrary waveform.


:AWG<x>:ARBitrary:SAVe:WAVEform 

Description Saves an arbitrary waveform.

Syntax :AWG<x>:ARBitrary:SAVe:WAVEform {ARB1 | ARB2 | ARB3 | ARB4 | <file path>}

Parameter	<x>	Channel number 1~2.
	ARB1~4	Saves the arbitrary waveform to one of the internal memory slots.
	<file path>	Saves the arbitrary waveform to disk or USB to the specified file path. Example: "Disk:/xxx.UAW" "USB:/xxx.UAW"

Example :AWG1:ARB:SAVE:WAVE ARB2
Saves the arbitrary waveform to ARB2.

:AWG<x>:ARBitrary:LOAd:WAVEform 

Description Loads an arbitrary waveform.

Syntax :AWG<x>:ARBitrary:LOAd:WAVEform {ARB1 | ARB2 | ARB3 | ARB4 | <file path>}

Parameter	<x>	Channel number 1~2.
	ARB1~4	Loads the arbitrary waveform from one of the internal memory slots.

<code><file path></code>	Loads the arbitrary waveform from disk or USB at the specified file path. Example: "Disk:/xxx.UAW" "USB:/xxx.UAW"
--------------------------------	--

Example `:AWG1:ARB:LOA:WAVE ARB2`
 Loads the arbitrary waveform from ARB2.

`:AWG<x>:ARBitrary:EDIT:COpy` Set

Description	Copies a segment of an arbitrary waveform to a specific point.
--------------------	--

Syntax	<code>:AWG<x>:ARBitrary:EDIT:COpy {<START> , <LENGth> , <PASTe>}</code>
---------------	---

Parameter	<code><x></code> Channel number 1~2.
	<code><START></code> NR1, point at which the segment to copy starts.
	<code><LENGth></code> NR1, length of the segment to copy.
	<code><PASTe></code> NR1, point at which the segment is to be copied.

Example `:AWG1:ARB:EDIT:COpy 5,100,106`
 Copies a segment of 100 points starting from point 5 of an arbitrary waveform and paste it to point 106 of this arbitrary waveform.

`:AWG<x>:ARBitrary:EDIT:CLear` Set

Description	Deletes a segment of an arbitrary waveform
--------------------	--

Syntax	<code>:AWG<x>:ARBitrary:EDIT:CLear { ALL <START> , <LENGth>}</code>
---------------	---

Parameter	<code><x></code> Channel number 1~2.
	<code>ALL</code> Deletes the entire arbitrary waveform.
	<code><START></code> NR1, point at which the segment to delete starts.

<LENGth> NR1, length of the segment to delete.

Example :AWG1:ARB:EDIT:CLE ALL

:AWG<x>:ARBitrary:EDIT:LINE 

Description Creates a line on an arbitrary waveform.

Syntax :AWG<x>:ARBitrary:EDIT:LINE {<address1> , <data1> , address2> , <data2>}

Parameter	<x>	Channel number 1~2.
	<address1>	NR1, the point at which the line starts.
	<data1>	NRf, the value at the starting point.
	<address2>	NR1, the point at which the line ends.
	<data2>	NRf, the value at the ending point.

Example :AWG1:ARB:EDIT:LIN 40,0.05,100,0.1
Creates a line between point 40 at value 0.05 and point 100 at value 0.01.

:AWG<x>:ARBitrary:EDIT:SCALE 

Description Sets the vertical scale of the arbitrary waveform.

Syntax :AWG<x>:ARBitrary:EDIT:SCALE {<NRf>}

Parameter	<x>	Channel number 1~2.
	<NRf>	Scale 0.1~ 10

Example :AWG1:ARB:EDIT:SCAL 5.5

:AWG<x>:ARBitrary:EDIT:POINT 

Description Edits a single point on an arbitrary waveform.

Syntax :AWG<x>:ARBitrary:EDIT:POINT {<address1> , <data1>}

Parameter	<x>	Channel number 1~2.
-----------	-----	---------------------

<code><address1></code>	NR1, the point to be edited.
<code><data1></code>	NRf, the value of that point.

Example :AWG1:ARB:EDIT:POIN 20,0.2

:AWG<x>:ARBitrary:EDIT:POINT:ADD 

Description Adds the edited point to the arbitrary waveform.

Syntax :AWG<x>:ARBitrary:EDIT:POINT:ADD {<NR1>}

Parameter	<code><x></code>	Channel number 1~2.
	<code><NR1></code>	The point to be added.

Example :AWG1:ARB:EDIT:POIN:ADD 20

:AWG<x>:ARBitrary:EDIT:POINT:DELEte 

Description Adds the edited point to the arbitrary waveform.



Syntax :AWG<x>:ARBitrary:EDIT:POINT:DELEte {<NR1>}


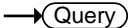
Parameter	<code><x></code>	Channel number 1~2.
	<code><NR1></code>	The point to be deleted.

Example :AWG1:ARB:EDIT:POIN:DELE 20

Data Logging Commands

:DATALOG:STATE	255
:DATALOG:SOURce	255
:DATALOG:SAVe	256
:DATALOG:INTerval	256
:DATALOG:DURation	257

		
		
<hr/>		
:DATALOG:STATE		
<hr/>		
Description	Sets or queries the state of the data logging app.	
Syntax	:DATALOG:STATE {OFF ON ?}	
Related commands	:DATALOG:SOURce :DATALOG:SAVe :DATALOG:INTerval :DATALOG:DURation	
Parameter/ Return parameter	OFF	Turns the data logging off.
	ON	Turns the data logging on.
Example	:DATALOG:STATE ON Turns the data logging app on.	

		
		
<hr/>		
:DATALOG:SOURce		
<hr/>		
Description	Sets or queries the data logging source channel.	
Syntax	:DATALOG:SOURce { CH1~CH4 D0~D15 all ? }	
Related commands	:DATALOG:STATE :DATALOG:SAVe :DATALOG:INTerval :DATALOG:DURation	
Parameter/Return	CH1 ~CH4	Channel 1, 2, 3 or 4

parameter	D0~D15	Digital channels D0~D15
	all	All displayed channels.

Example :DATALOG:SOURce CH1
Sets the source to CH1.

Set →

→ Query

:DATALOG:SAVE

Description Sets or queries the save format as image or waveform.

Syntax :DATALOG:SAVE {IMAGE|WAVEform|?}

Related commands :DATALOG:STATE
:DATALOG:SOURce
:DATALOG:INTerval
:DATALOG:DURation

Parameter/Return parameter	IMAGE	Save as images.
	WAVEform	Save as waveforms.

Example :DATALOG:SAVE WAVEform
Sets the save format to waveform.

Set →

→ Query

:DATALOG:INTerval


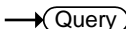
Description Sets or queries the interval time between each recording.

Syntax :DATALOG:INTerval <NRF>
:DATALOG:INTerval?

Related commands DATALOG:SAVE

Parameter/Return parameter	<NRF>	Discrete time intervals in seconds.
----------------------------	-------	-------------------------------------

Example :DATALOG:INT 2
Sets the interval time to 2 seconds

:DATALOG:DURation		 
Description	Sets or queries the duration time of each recording.	
Syntax	:DATALOG:DURation <NRf> :DATALOG:DURation?	
Parameter/Return parameter	<NRf>	Discrete recording time in seconds.
Example	:DATALOG:DUR 5 Sets the recording time to 5 seconds.	

Remote Disk Commands

:REMOTEDisk:IPADdress	258
:REMOTEDisk:PATHName	258
:REMOTEDisk:USERName	259
:REMOTEDisk:PASSWord	259
:REMOTEDisk:MOUNT	259
:REMOTEDisk:AUTOMount	260

:REMOTEDisk:IPADdress
 Set →
 Query

Description	Sets or returns the IP address of remote disk.	
Syntax	:REMOTEDisk:IPADdress {<string> ?}	
Parameter/ Return parameter	<string>	IP address enclosed in double quotes. Eg., 172.16.20.255
Example	:REMOTEDisk:IPADdress "172.16.20.255" Sets the remote disk IP address as 172.16.20.255.	

:REMOTEDisk:PATHName
 Set →
 Query

Description	Sets or returns the file path of the remote disk.	
Syntax	:REMOTEDisk:PATHName {<string> ?}	
Parameter/ Return parameter	<string>	File path in enclosed in double quotes eg., "remote_disk"
Example	:REMOTEDisk:PATHName "remote_disk" Sets the file path to c:/remote_disk.	

Set →
 → Query

:REMOTEDisk:USERName

Description	Sets or queries the account username for the remote disk.	
Syntax	:REMOTEDisk:USERName {<string> ? }	
Parameter/Return parameter	<string>	User name enclosed in double quotes eg., "User_Name".
Example	:REMOTEDisk:USERName "User_Name" Sets the account name as User_Name.	

Set →
 → Query

:REMOTEDisk:PASSWORD

Description	Sets or queries the account password for the remote disk.	
Syntax	:REMOTEDisk:PASSWORD {<string> ? }	
Parameter/Return parameter	<string>	Username password enclosed in double quotes eg., "Password".
Example	:REMOTEDisk:PASSWORD "Password" Sets the account password as Password.	

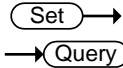
Set →
 → Query

:REMOTEDisk:MOUNT

Description	Turns remote disk on/off or queries its state.	
Syntax	:REMOTEDisk:MOUNT { OFF ON ? }	
Parameter/Return parameter	OFF	Unmount remote disk
	ON	Mount remote disk

Example :REMOTEDisk:IPADdress "172.16.5.154"
 :REMOTEDisk:PATHName "remote_disk"
 :REMOTEDisk:USERName "guest"
 :REMOTEDisk:PASSWord "password"
 :REMOTEDisk:MOUNT ON

Sets the remote disk parameters and mounts the remote disk.



:REMOTEDisk:AUTOMount

Description Turns automount on/off or queries its state. The remote disk must be configured beforehand.

Syntax :REMOTEDisk:AUTOMount { OFF | ON | ? }

Parameter/Return parameter	OFF	Don't mount the remote disk at start up.
	ON	Automatically mount the remote disk on start up.

Example :REMOTEDisk:AUTOMount ON

Turns the automount function on.

DMM Commands (For the MDO-2000EX series only)

:DMM.....	261
:DMM:STATE.....	261
:DMM:VALue.....	262
:DMM:HOLD.....	262
:DMM:MMIN.....	262
:DMM:MODE.....	263
:DMM:MODE:RANGe.....	264
:DMM:TEMPerature:UNITs.....	264
:DMM:TEMPerature:TYPE.....	265
:DMM:TEMPerature:SIM.....	265

:DMM → Query

Description	Returns the DMM status.	
Syntax	:DMM?	
Related commands	:MEASUrement:DISPlay	
Parameter/Return parameter	<string>	Returns the mode, current measurement, max measurement, minimum measurement, Hold state.
Example	:DMM? Mode:ACV,Value:0.000,Max Value:0.000,Min Value:0.000,Hold:OFF	

:DMM:STATE Set →
→ Query

Description	Sets or queries the DMM function .
Syntax	:DMM: STATE { ON OFF ?}

Parameter/Return parameter	ON	Turns the DMM function on.
	OFF	Turns the DMM function off.

Example :DMM:STATE ON
Turns the DMM function on.

:DMM:VALue → Query

Description	Returns the measurement value.	
Syntax	:DMM:VALue?	
Related commands	:MEASUrement:DISPlay	
Return parameter	<string>	Returns the measurement or value on the display as a string.

Example :DMM:VALue?
0.000
Returns the value on the DMM display.

Set →

:DMM:HOLD → Query

Description	Sets or queries the Hold function status.	
Syntax	:DMM:HOLD { ON OFF ?}	
Parameter/Return parameter	ON	Turns the Hold function on.
	OFF	Turns the Hold function off.

Example :DMM:HOLD ON
Turns the Hold function on.

Set →

:DMM:MMIN → Query

Description	Sets or queries the maximum and minimum status.	
Syntax	:DMM: MMIN { ON OFF ?}	

Parameter/Return parameter	ON	Turns the maximum and minimum function on.
	OFF	Turns the maximum and minimum function off.

Example :DMM:MMIN ON
Turns the maximum and minimum function on.

:DMM:MODE 

Description Sets or queries the DMM mode.

Syntax :DMM:MODE
{ DCV | DCMV | ACV | ACMV | DCA | DCMA | ACA | ACMA | OHM | DIODE | BEEP | TEMPerature | ? }

Parameter/Return parameter	DCV	DCV mode
	DCMV	DCMV mode
	ACV	ACV mode
	ACMV	ACMV mode
	DCA	DCA mode
	DCMA	DCMA mode
	ACA	ACA mode
	ACMA	ACMA mode
	OHM	Resistance measurement mode
	DIODE	Diode tester
	BEEP	Continuity tester
	TEMPerature	Temperature measurement mode

Example :DMM:MODE DCV
Sets the measurement mode to DCV.

Set →

→ Query

:DMM:MODE:RANGe

Description	Sets or queries the DMM measurement range.	
Syntax	:DMM:MODE:RANGe (AUTo <NRf>) :DMM:MODE:RANGe?	
Related commands	:DMM:MODE	
Parameter/Return parameter	AUTo <NRf>	Auto range ACV: 5,50,750 DCV: 5,50,500,1000 ACmV: 0.5,0.05 DCmV: 0.5,0.05 ACmA: 0.5,0.05 DCmA: 0.5,0.05 ACA: 10 DCA: 10

Example :DMM:MODE ACV
:DMM:MODE:RANGe AUTo
Sets the ACV measurement to auto range.

Set →

→ Query

:DMM:TEMPerature:UNITs

Description	Sets the units for the temperature measurement function.	
Syntax	:DMM:TEMPerature:UNITs { Celsius Fahrenheit ? }	
Parameter/Return parameter	Celsius Fahrenheit	Degrees Celsius Degrees Fahrenheit

Example :DMM:TEMPerature:TYPE Celsius
Sets the temperature measurement to °C.

:DMM:TEMPerature:TYPe




Description	Sets the type of thermocouple used for the temperature measurement function.	
Syntax	:DMM:TEMPerature:TYPe { TYPEB TYPEE TYPEJ TYPEK TYPEN TYPER TYPES TYPET ? }	
Parameter/Return parameter	TYPEB	B
	TYPEE	E
	TYPEJ	J
	TYPEK	K
	TYPEN	N
	TYPER	R
	TYPES	S
	TYPET	T

Example :DMM:TEMPerature:TYPe K
 Sets the temperature measurement function to use the K type thermocouple.

:DMM:TEMPerature:SIM




Description	Set or returns the environment temperature when temperature measurement selected.	
Syntax	:DMM:TEMPerature:SIM {<NRf>} :DMM:TEMPerature:SIM?	
Related commands	:DMM:MODE :DMM:TEMPerature:UNITs	
Parameter/Return parameter	<NRf>	0.0~50.0 for Celsius degrees; 32~ 122.0 for Fahrenheit degrees

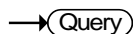
Example :DMM:MODE TEMPerature
 :DMM:TEMPerature:UNITs Celsius
 :DMM:TEMPerature:SIM 23.5

Sets the environment temperature to 23.5 Celsius degrees.

Spectrum Analyzer Commands

:SA:MEMory?	267
:SA:MEMory:SOURce	270
:SA:STATE	270
:SA:SOURce	270
:SA:SPECTRUMTrace	271
:SElect:NORMAL	271
:SElect:MAXHold	271
:SElect:MINHold	272
:SElect:AVErage	272
:SA:AVErage:NUMAVg	273
:SA:DETECTionmethod:MODE	273
:SA:DETECTionmethod:MAXHold	273
:SA:DETECTionmethod:MINHold	274
:SA:DETECTionmethod:NORMAL	274
:SA:DETECTionmethod:AVErage	275
:SA:FREQuency	275
:SA:SPAN	276
:SA:START	276
:SA:STOP	276
:SA:RBW:MODE	277
:SA:RBW	277
:SA:SPANRbwratio	278
:SA:WINDow	278
:SA:UNIts	279
:SA:SCALE	279
:SA:POSition	280

:SA:MEMory?



Description	Returns the data in acquisition memory for the spectrum analyzer function as a header + raw data.
-------------	---

Syntax :SA:MEMory?

Related Commands :SA:MEMory:SOURce

Return parameter <string> <waveform block data>

Returns acquisition settings followed by raw waveform block data.

<string>

Returns the spectrum analyzer settings .

Format:

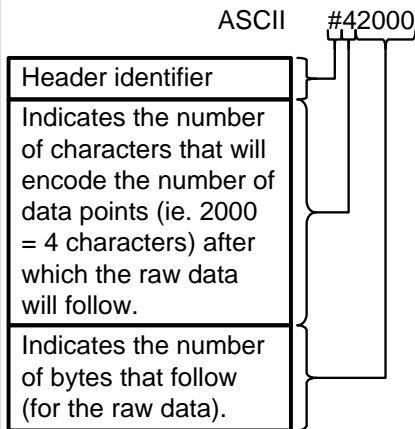
parameter(1),setting(1);parameter(2),setting(2)...parameter(n),setting(n);Waveform Data;

<waveform block data>

Header followed by the raw waveform data.

Format:

Header: The header (in ASCII) encodes the number of bytes for the header followed by the number of data points in bytes for the raw data.



Raw Data:

Each two bytes (in hex) encodes the vertical data of a data point. The data

is signed hex data (2's complement, -32768 ~ 32767).

Waveform Raw Data Example:

Header raw data.....

Hex:

23 34 32 30 30 30 00 1C 00 1B 00 1A 00
1A 00 1B

ASCII/Decimal:

#42000 28 27 26 26 27.....

The actual value of a data point can be calculated with the following formula:
(Decimal value of hex data/ vertical division Factor) * vertical scale.

Note: Vertical division Factor is fixed as 25. The vertical scale is returned with the acquisition settings that precede the raw data.

For example if the raw data for a point is 001C (=28 decimal) then,
 $(28/25) \times 10\text{dB} = 11.2\text{dB}$

Example

:SA:MEMory?

Format,2.0E;Firmware,V1.28;Time,24-Apr-17
15:54:49;Memory
Length,1.000E+03;Source,CH1;Probe
Ratio,1.000E+00;Vertical Unit,dB;Vertical
Position,3.000E+00;Vertical
Scale,2.000E+01;Horizontal Unit,Hz;Horizontal
Scale,1.000E+04;Sampling Period,1.000E+02;Center
Frequency,2.300E+03;Span,1.000E+05;FREQUENC
Y,NORM,Waveform Data;
#42000 follows waveform block
data in hex

Note

You need to set the command “:USBDelay ON” once to avoid missing data before running the program if you use USB interface to transfer data in the Windows 10 operating system.

Set →
→ Query

:SA:MEMory:SOURce

Description	Sets or returns the source of the waveform data	
Syntax	:SA:MEMory:SOURce {NORMal AVErage MAXHold MINHold} :SA:MEMory:SOURce?	
Parameter/Return parameter	NORMal	Normal data
	AVErage	Average data
	MAXHold	Maxhold data
	MINHold	Minhold data
Example	:SA:MEMory:SOURce AVE Sets the memory source to average data	

Set →
→ Query

:SA:STATE

Description	Sets or returns the state of spectrum analyzer.	
Syntax	:SA:STATE {ON OFF}	
Parameter/Return parameter	ON	Turns the spectrum analyzer on.
	OFF	Turns the spectrum analyzer off
Example	:SA:STATE ON Sets the spectrum analyzer on.	

Set →
→ Query

:SA:SOURce

Description	Sets or returns the source of the spectrum analyzer	
Syntax	:SA:SOURce {CH1 CH2 CH3 CH4} :SA:SOURce?	
Parameter/Return parameter	CH1	Channel one
	CH2e	Chnanel two

	CH3	Channel three
	CH4	Channel four

Example :SA:SOURce CH2
 Sets the source of spectrum analyzer to channel two

:SA:SPECTRUMTrace (Set) →

Description	Resets all spectrum traces.	
Syntax	SA:SPECTRUMTrace {RESET}	
Parameter	RESET	Reset the trace
Example	:SA:SPECTRUMTrace RESET Reset all the trace of spectrum analyzer	

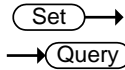
:SElect:NORMAL (Set) →
→ (Query)

Description	Sets or returns the frequency domain Normal trace display on or off in the frequency domain graticule.	
Syntax	:SElect:NORMAL {ON OFF} :SElect:NORMAL?	
Parameter/Return parameter	ON	Turns the normal trace display on.
	OFF	Turns the normal trace display off.
Example	:SElect:NORMAL ON Sets the normal trace display on	

:SElect:MAXHold (Set) →
→ (Query)

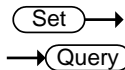
Description	Sets or returns the frequency domain Max Hold trace display on or off in the frequency domain graticule.	
--------------------	--	--

Syntax	:SElect:MAXHold {ON OFF}	
	:SElect:MAXHold?	
Parameter/Return parameter	ON	Turns the Max Hold trace display on.
	OFF	Turns the Max Hold trace display off.
Example	:SElect:MAXHold OFF	
	Sets the Max Hold trace display off	



:SElect:MINHold

Description	Sets or returns the frequency domain Min Hold trace display on or off in the frequency domain graticule.	
Syntax	:SElect:MINHold {ON OFF}	
	:SElect:MINHold?	
Parameter/Return parameter	ON	Turns the Min Hold trace display on.
	OFF	Turns the Min Hold trace display off.
Example	: SElect:MINHold OFF	
	Sets the Min Hold trace display off	



:SElect:AVERage

Description	Sets or returns the frequency domain Average trace display on or off in the frequency domain graticule.	
Syntax	:SElect:AVERage {ON OFF}	
	:SElect: AVERage?	
Parameter/Return parameter	ON	Turns the Average trace display on.
	OFF	Turns the Average trace display off.
Example	: SElect: AVERage ON	
	Sets the Average trace display on	

:SA:AVERage:NUMAVg



Description	Sets or returns the number of acquisitions to be used when creating the Average frequency domain trace.	
Syntax	:SA:AVERage:NUMAVg {<NR1>} :SA:AVERage:NUMAVg?	
Parameter/Return parameter	<NR1>	The range is 2 - 256, in exponential increments.
Example	:SA:AVERage:NUMAVg 128 Sets the Average number to 128	

:SA:DETECTionmethod:MODE



Description	Sets or returns the detection within the oscilloscope occurs automatically or manually.	
Syntax	:SA:DETECTionmethod:MODE {AUTo MANual} :SA:DETECTionmethod:MODE?	
Related commands	:SA:DETECTionmethod:MAXHold,:SA:DETECTionmethod:MINHold :SA:DETECTionmethod:NORMal,:SA:DETECTionmethod:AVERage	
Parameter/Return parameter	AUTo	Automatically mode
	MANual	Manually mode
Example	:SA:DETECTionmethod:MODE AUTo Sets the detection mode to automatic.	

:SA:DETECTionmethod:MAXHold

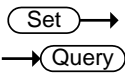


Description	Sets or returns the detection method of max Hold frequency domain trace.	
-------------	--	--

Syntax :SA:DETECTIonmethod:MAXHold
 {PLUSpeak | MINUSpeak | SAMPlE | AVERage}
 :SA:DETECTIonmethod:MAXHold?

Parameter/Return parameter	PLUSpeak	Sets the detection method to plus peak.
	MINUpeak	Sets the detection method to minus peak
	SAMple	Sets the detection method to sample.
	AVERage	Sets the detection method to average.

Example :SA:DETECTIonmethod:MAXHold AVERage
 Sets the detection method to average.



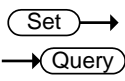
:SA:DETECTIonmethod:MINHold

Description Sets or returns the detection method of min Hold frequency domain trace.

Syntax :SA:DETECTIonmethod:MINHold
 {PLUSpeak | MINUSpeak | SAMPlE | AVERage}
 :SA:DETECTIonmethod:MINHold?

Parameter/Return parameter	PLUSpeak	Sets the detection method to plus peak.
	MINUpeak	Sets the detection method to minus peak
	SAMple	Sets the detection method to sample.
	AVERage	Sets the detection method to average.

Example :SA:DETECTIonmethod:MINHold AVERage
 Sets the detection method to average.



:SA:DETECTIonmethod:NORMAL

Description Sets or returns the detection method of normal frequency domain trace.

Syntax :SA:DETECTIonmethod:NORMAl
 {PLUSpeak | MINUSpeak | SAMPlE | AVErage}
 :SA:DETECTIonmethod:NORMAl?

Parameter/Return parameter	PLUSpeak	Sets the detection method to plus peak.
	MINUpeak	Sets the detection method to minus peak
	SAMple	Sets the detection method to sample.
	AVErage	Sets the detection method to average.

Example :SA:DETECTIonmethod:NORMAl AVErage
 Sets the detection method to average.

:SA:DETECTIonmethod:AVErage  

Description Sets or returns the detection method of average frequency domain trace.

Syntax :SA:DETECTIonmethod:AVErage
 {PLUSpeak | MINUSpeak | SAMPlE | AVErage}
 :SA:DETECTIonmethod:AVErage?

Parameter/Return parameter	PLUSpeak	Sets the detection method to plus peak.
	MINUpeak	Sets the detection method to minus peak
	SAMple	Sets the detection method to sample.
	AVErage	Sets the detection method to average.

Example :SA:DETECTIonmethod:AVErage AVErage
 Sets the detection method to average.

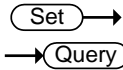
:SA:FREQUency  

Description Sets or returns the frequency (or center frequency) of the acquisition system.

Syntax :SA:FREQuency {<NRf> | CENTER}
:SA:FREQuency?

Parameter/Return parameter	<NRf>	Sets the frequency by user.
	CENTER	Sets the frequency of Max. peak marker to center.

Example SA:FREQuency 3.0E+06
sets the center frequency to 3 MHz.



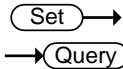
:SA:SPAN

Description Sets or returns the span frequency setting.

Syntax :SA:SPAN <NRf>
:SA:SPAN?

Parameter/Return parameter	<NRf>	Sets the span frequency by user.
----------------------------	--------------------	----------------------------------

Example SA:SPAN 25E+06
sets the span frequency to 25 MHz



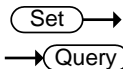
:SA:START

Description Sets or returns the start frequency setting.

Syntax :SA:START <NRf>
:SA:START?

Parameter/Return parameter	<NRf>	Sets the start frequency by user.
----------------------------	--------------------	-----------------------------------

Example SA:START -9.5E+06
sets the start frequency to -9.5 MHz.



:SA:STOP

Description Sets or returns the stop frequency setting.

Syntax :SA:STOP <NRf>
:SA:STOP?

Parameter/Return parameter <NRf> Sets the stop frequency by user.

Example SA:START 100E+06
sets the stop frequency to 100MHz.

Set →

→ Query

:SA:RBW:MODE

Description Sets or returns the resolution bandwidth (RBW) mode, either automatic or manual.

Syntax :SA:RBW:MODE {AUTo | MANual}
:SA:RBW:MODE?

Parameter/Return parameter AUTo Automatically mode.
MANual Manually mode.

Example SA:RBW:MODE AUTo
sets the mode to automatic.

Set →

→ Query

:SA:RBW

Description Sets or returns the resolution bandwidth (RBW) when the RBW mode has been set to MANUAL (using the command SA:RBW:MODE).

Syntax :SA:RBW <NRf>
:SA:RBW?

Related commands SA:RBW:MODE

Parameter/Return parameter <NRf> Sets the RBW by user.

Example Sets SA:RBW 2.0E+04
 Query SA:RBW?
 Return 1.825017e+04

If the RBW set to 20kHz, the query will return the nearest value (1.825017e+04).

Set →

→ Query

:SA:SPANRbwratio

Description Sets or returns the resolution bandwidth (RBW) when the RBW mode has been set to AUTO (using the command SA:RBW:MODE).

Syntax :SA:SPANRbwratio
 {RATIO1K | RATIO2K | RATIO5K | <NRF>}
 :SA:SPANRbwratio?

Related commands SA:RBW:MODE

Parameter/ Return parameter	<NRF>	Sets the RBW by user.
	RATIO1K	1000 : 1
	RATIO2K	2000 : 1
	RATIO5K	5000 : 1

Example :SA:SPANRbwratio RATIO2K
 Sets the ratio to 2000:1.

Sets :SA:SPANRbwratio 2000

Query :SA:SPANRbwratio?

Return RATIO2K

Set →

→ Query

:SA:WINDow

Description Sets or returns the windowing function, which is only used for traces.

Syntax	:SA:WINDow {RECTangular HAMming HANning BLAckman}:SA:WINDow?	
Parameter/ Return parameter	RECTangular	Sets to Rectangular window
	HAMming	Sets to Hamming window
	HANning	Sets to Hanning window
	BLAckman	Sets to Blackman window
Example	:SA:WINDow HANning Sets to the hanning window.	

Set →

→ Query

:SA:UNIts

Description	Sets or returns the vertical units.	
Syntax	:SA:UNIts {DBV LINEAR DBM} :SA:UNIts?	
Parameter/ Return parameter	DBV	Sets to DBV unit
	LINEAR	Sets to Linear unit
	DBM	Sets to DBM unit
Example	:SA:UNIts DBM Sets the unit to DBM unit.	

Set →

→ Query

:SA:SCALE

Description	Sets or returns the overall vertical scale.	
Syntax	:SA:SCALE <NRf> :SA:SCALE?	
Related commands	:SA:UNIts	

Parameter/ Return parameter	<NRf>	Vertical scale, the value may vary which depends on the unit selected. dBm and dBV : 1, 2, 5, 10, 20 (dB) Linear: 2m, 5m, 10m, 20m, 50m, 100m, 200m, 500m, 1, 2, 5, 10, 20, 50, 100, 200, 500, 1k (V)
--------------------------------	-------	---

Example :SA:SCAlE 2
Sets the scale to 2.

:SA:POSition (Set) →
← (Query)

Description	Sets or returns the overall vertical position.	
Syntax	:SA:POSition <NRf> :SA:POSition?	

Parameter/ Return parameter	<NRf>	Vertical position range: +/-12
--------------------------------	-------	--------------------------------

Example :SA:POSition 3
Sets the vertical position to 3.

Power Supply Commands (For the MDO-2000EX series only)


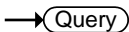
:POWERSupply:OUTPut<X>	281
:POWERSupply:OUTPut<X>:VOLTage	281
:POWERSupply:OUTPut<X>:RECONFigure	282
:POWERSupply:OUTPut<X>:OCP	282
:POWERSupply:CONFigure	283

:POWERSupply:OUTPut<X>
 Set →
 → Query


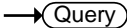
Description	Sets or returns the power supply output.	
Syntax	:POWERSupply:OUTPut<X> {ON OFF} :POWERSupply:OUTPut<X>?	
Parameter/ Return parameter	OFF ON <X>	Turns off the power supply output Turns on the power supply output Range 1~2, Select the output 1 or output 2
Example	:POWERSupply:OUTPut1 ON Turn on output 1.	
Note	Please run the command “:POWERSupply:CONFigure ON” once before setting for the first time.	

:POWERSupply:OUTPut<X>:VOLTage
 Set →
 → Query

Description	Sets or returns the power supply voltage.	
Syntax	:POWERSupply:OUTPut<X>:VOLTage <NR3> :POWERSupply:OUTPut<X>:VOLTage?	
Parameter/	<NR3>	Range 1.0~5.0, sets the voltage

		 
:POWERSupply:CONFigure		
Description	Configure the power supply function. *It takes 6 seconds to perform the initialization setting.	
Syntax	:POWERSupply:CONFigure{ON ?}	
Parameter/ Return parameter	OFF ?	Con figure the power supply function. Qurey the configuration of power supply.
Example	:POWERSupply:CONFigure ON Configure the power supply function.	
Note	It is necessary to run this command once to initialize settings after the device is turned on.	

USB Delay Command

:USBDelay	 				
Description	Sets or returns the USB delay function for the PC connection which Windows 10 installed				
Syntax	:USBDelay {OFF ON} :USBDelay?				
Parameter/ Return parameter	<table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 15%; padding: 2px;"><ON></td> <td style="padding: 2px;">Turns on the USB delay function</td> </tr> <tr> <td style="width: 15%; padding: 2px;"><OFF></td> <td style="padding: 2px;">Turns off the USB delay function</td> </tr> </table>	<ON>	Turns on the USB delay function	<OFF>	Turns off the USB delay function
<ON>	Turns on the USB delay function				
<OFF>	Turns off the USB delay function				
Example	:USBDelay ON Turns on the USB delay function when the scope connected with window 10 installed PC.				
Note 1	This command needs to be runned once only when you use USB interface to transfor remote command in the Windiwos 10 operating system.				
Note 2	It is known that there is a problem with missing data when using the USB CDC ACM protocol in the Windows 10 operationg system to receive massive data from the oscilloscope (raw data / image data). This problem can be solved by reducing the data output rate at the oscilloscope. The same problem doesn't occur on other versions of Windows operating systems.				

APPENDIX

Error messages

Description The following error messages may be returned from the :SYSTem:ERRor? query. For details see page 157.

List of error messages

Error number, "Error Description"

+0, "No error."
 -100, "Command error"
 -101, "Invalid character"
 -102, "Syntax error"
 -103, "Invalid separator"
 -104, "Data type error"
 -105, "GET not allowed"
 -108, "Parameter not allowed"
 -109, "Missing parameter"
 -110, "Command header error"
 -111, "Header separator error"
 -112, "Program mnemonic too long"
 -113, "Undefined header"
 -114, "Header suffix out of range"
 -115, "Unexpected number of parameters"
 -120, "Numeric data error"
 -121, "Invalid character in number"
 -123, "Exponent too large"
 -124, "Too many digits"
 -128, "Numeric data not allowed"
 -130, "Suffix error"
 -131, "Invalid suffix"
 -134, "Suffix too long"
 -138, "Suffix not allowed"

- 140, "Character data error"
- 141, "Invalid character data"
- 144, "Character data too long"
- 148, "Character data not allowed"
- 150, "String data error"
- 151, "Invalid string data"
- 158, "String data not allowed"
- 160, "Block data error"
- 161, "Invalid block data"
- 168, "Block data not allowed"
- 170, "Expression error"
- 171, "Invalid expression"
- 178, "Expression data not allowed"
- 180, "Macro error"
- 181, "Invalid outside macro definition"
- 183, "Invalid inside macro definition"
- 184, "Macro parameter error"

- 200, "Execution error"
- 201, "Invalid while in local"
- 202, "Settings lost due to rtl"
- 203, "Command protected"
- 210, "Trigger error"
- 211, "Trigger ignored"
- 212, "Arm ignored"
- 213, "Init ignored"
- 214, "Trigger deadlock"
- 215, "Arm deadlock"
- 220, "Parameter error"
- 221, "Settings conflict"
- 222, "Data out of range"
- 223, "Too much data"
- 224, "Illegal parameter value"
- 225, "Out of memory"
- 226, "Lists not same length"
- 230, "Data corrupt or stale"
- 231, "Data questionable"
- 232, "Invalid format"
- 233, "Invalid version"
- 240, "Hardware error"

- 241, "Hardware missing"
- 250, "Mass storage error"
- 251, "Missing mass storage"
- 252, "Missing media"
- 253, "Corrupt media"
- 254, "Media full"
- 255, "Directory full"
- 256, "File name not found"
- 257, "File name error"
- 258, "Media protected"
- 260, "Expression error"
- 261, "Math error in expression"
- 270, "Macro error"
- 271, "Macro syntax error"
- 272, "Macro execution error"
- 273, "Illegal macro label"
- 274, "Macro parameter error"
- 275, "Macro definition too long"
- 276, "Macro recursion error"
- 277, "Macro redefinition not allowed"
- 278, "Macro header not found"
- 280, "Program error"
- 281, "Cannot create program"
- 282, "Illegal program name"
- 283, "Illegal variable name"
- 284, "Program currently running"
- 285, "Program syntax error"
- 286, "Program runtime error"
- 290, "Memory use error"
- 291, "Out of memory"
- 292, "Referenced name does not exist"
- 293, "Referenced name already exists"
- 294, "Incompatible type"

- 300, "Device-specific error"
- 310, "System error"
- 311, "Memory error"
- 312, "PUD memory lost"
- 313, "Calibration memory lost"
- 314, "Save/recall memory lost"

- 315, "Configuration memory lost"
- 320, "Storage fault"
- 321, "Out of memory"
- 330, "Self-test failed"
- 340, "Calibration failed"
- 350, "Queue overflow"
- 360, "Communication error"
- 361, "Parity error in program message"
- 362, "Framing error in program message"
- 363, "Input buffer overrun"
- 365, "Time out error"

- 400, "Query error"
- 410, "Query INTERRUPTED"
- 420, "Query UNTERMINATED"
- 430, "Query DEADLOCKED"
- 440, "Query UNTERMINATED after indefinite response"

INDEX

ACQuire	
FILTer	
TRACking	42
ACQuire	
AVERAge	38
FILTer	
FREQUency	42
FILTer.....	42
SOURce	41
MEMory.....	39
MODE.....	39
ACQuire	
STATe	43
ACQuire	
INTERpolation.....	43
ACQuire	
RECOrdlength	43
ACQuire	
HEADer	44
ACQuire	
MEMory.....	267
AUTORSET	
MODE.....	45
AUTOSet.....	45
BUS1	165
CAN	
BITRate	174
PROBe.....	173
SAMPLepoint.....	174
SOURce.....	173
DISPlay	
FORMAt	172
I2C	
ADDRess	
RWINClude.....	166
SCLK	
SOURce	167
SDA	
SOURce	167
INPut	166
LABEL.....	216
DISPlay	216
LIN	
BITRate	175
IDFORmat	175
POLARity	175
SAMPLepoint.....	176
SOURce.....	176
STANDard	176
SPI	
BITORder	171
MISO	
SOURce	172
MOSI	
SOURce	172
SCLK	

POLARity	170	DDT	64
SOURce	171	H1Position	65
SS		H2Position	65
POLARity	170	HDELta	66
SOURce	171	HUNI.....	63
WORDSize	170	HUSE.....	63
STATE.....	165	MODE.....	62
TYPE.....	165	SOURce	62
UART		V1Position.....	66
BITRate	167	VDELta.....	67
DATABits.....	168	VUNI.....	64
EOFPacket.....	169	VUSE	64
PACKEt	168	XY	
PARItY	168	POLar	
RX		RADIUS	
SOURce.....	169	DELta	69
TX		POSition.....	68
SOURce.....	169	THETA	
CHANnel		DELta	69
BWLimit	46	POSition.....	69
COUPLing.....	47	PRODUct	
DESKew.....	47	DELta	70
DISPlay	47	POSition.....	70
EXPand	48	RATio	
IMPedance	48	DELta	71
INVert.....	49	RATio	
LABel	213	POSition.....	70
DISPlay.....	214	RECTangular	
POSition.....	49	X	
PROBe		DELta	67
TYPE.....	50	POSition.....	67
PROBe		Y	
RATio	50	DELta	68
SCALe.....	50	POSition.....	68
CLS.....	34	DATALOG	
CURSor		DURATION.....	257, 273, 274, 275, 276, 277

INTErval.....	256	Error list	285
SAVe	256	Ethernet	
SOURce.....	255	interface.....	5
STATe.....	255	ETHERnet	
DATE	163	DHCP	163
DISPlay		FORCe	116
INTENSITy		GONogo	
BACKLight		CLEar.....	230
AUTODim		EXECute.....	230, 237
TIME	73	FUNCTion.....	231, 237
DISPlay		NGCount.....	231, 238, 239, 240, 241,
GRATicule.....	74	242, 243, 244, 245, 246, 247, 248,	
INTensity		249, 250, 251, 252, 253, 278, 279,	
BACKLight	73	280, 281, 282, 283, 284	
BACKLight		NGDefine	231
AUTODim		SCRipt.....	232
ENABle.....	73	SOURce	232
GRATicule	72	VIOLation	232
WAVEform	72	HARDcopy	
OUTPut.....	75	ASSIGN	78
PERSistence	74	MODE.....	76
WAVEform.....	75	PRINTINKSaver.....	77
DMM		SAVEFORMat.....	77
HOLD.....	261, 262	SAVEINKSaver	77
MOD	263	START	76
TEMPerature		HEADER.....	44
TYPe	265, 270, 271, 272	IDN?.....	32
UNITs.....	264	Interface.....	3
VALue.....	262, 264	LISTer	
DVM		DATA	173
MODE.....	229	LRN?	32
SOURce.....	228		
STATe.....	228		
VALue.....	229		

MARK.....	178	FPReshoot.....	84
CREATE.....	178	FREQuency.....	84
DELEte.....	179	FRFDelay.....	98
MATH		FRRDelay.....	98
ADVanced		GATing.....	80
OPERator.....	57	HIGH.....	93
POSition.....	59	LFFDelay.....	103
SCALe.....	59	LFRDelay.....	102
DISP.....	52	LOW.....	93
DUAL		LRFDelay.....	101
OPERator.....	53	LRRDelay.....	100, 101
POSition.....	54	MAX.....	94
SCALe.....	54	MEAN.....	91
SOURce.....	53	METHod.....	81
FFT		MIN.....	94
HORizontal		NEDGE.....	90
POSition.....	57	NPULSE.....	89
SCALe.....	57	NWIDth.....	85
MAG.....	55	PDUTy.....	85
POSition.....	56	PEDGE.....	90
SCALe.....	56	PERiod.....	86
SOURce.....	55	PHAsE.....	103, 104
WINDow.....	55	PK2PK.....	95
TYPE.....	53	PPULSE.....	89
MATHVAR.....	58	PWIDth.....	86
MEASure		RISe.....	87
AMPlitude.....	91	RMS.....	95
AREa.....	97	ROVShoot.....	87
CAReA.....	97	RPReshoot.....	88
CMEan.....	92	SOURce.....	80
CRMS.....	96		
FALL.....	83	MEASUREMENT	
FFFDelay.....	100	MEAS	
FFRDelay.....	99	MAXimum.....	108
FOVShoot.....	83	MEAN.....	109
		MINimum.....	109

SOURCE.....	105	USERName	259
STATE	106	RST	34
STDdev.....	110	RUN	115
TYPe	106	SAV	33
VALue	107	SAVE	
REFLevel		IMAGe.....	159
PERCent		FILEFormat	159
HIGH	81	INKSaver	160
LOW	82	SETUp	160
MID	82	WAVEform	161
MID2	82	FILEFormat	162
STATIstics.....	111	SEARCH	
MODE.....	111	COPY	181
WEIghting.....	111	FFTPeak	
RCL.....	33	METHod.....	210
RECALL		MPEak	211
SETUp	158	SINFo	211
WAVEform.....	158	STATE	182
REF		TOTAL	182
DISPlay	112	TRIGger	
LABEL	214	BUS	
DISPlay.....	215	B1	
OFFSet.....	113	CAN	
SCALE.....	114	CONDition	201
TIMEbase		DATA	
POStion	112	QUALifier	204
SCALE.....	113	SIZe	205
Remote control		VALue	205
interface configuration.....	3	FRAMEtype.....	202
REMOTEDisk		Identifier	
AUTOMount.....	260	DIRection	203
IPADDress.....	258	MODE	202
MOUNT.....	259	VALue	203
PASSWord.....	259	I2C	
PATHName.....	258	ADDRESS	

DIRection	193	POLarity	185
MODE	191	RISEFall	
TYPe	191	SLOP	186
VALue	192	TIME	189
CONDition	190	WHEn	189
DATA		RUNT	
SIZE	193	POLarity	186
VALue	194	TIME	188
LIN		WHEn	188
CONDition	206	SOURce	183
DATA		TYPe	182
QUALifier	207	SEGMENTS	
SIZE	208	CURRent	220
VALue	208	DISPALL	221
ERRTYPE	209	MEASURE	
IDENTifier		MODE	221
VALue	209	PLOT	
SPI		DIVide	222
CONDition	198	RESults	223
DATA		SElect	223
MISO	199	SOURce	222
MOSI	200	TABLE	
SIZE	199	LIST	225
UART		SAVE	225
CONDition	195	SElect	224
RX		SOURce	224
DATA	196	SAVE	225
TX		SElect	
DATA	197	END	227
TYPe	190	START	226
EDGE		SOURce	226
SLOP	183	STATE	219
HLEVel	184	TIME	221
LEVel	184	TOTalnum	220
LLEVel	185	SET	
PULSE		LABel	217
TIME	187	SINGLE	115
WHEn	187		
PULSEWidth			

Socket server	CONDition 148
function check.....8	DATA
Socket server	QUALifier 150
interface7	SIZE 151
STOP..... 115	VALue..... 152
SYSTEM	FRAMeType 148
ERRor.....157	Identifier
LOCK.....157	DIRection..... 150
TEMPlate	MODE.....149
MAXimum233	VALue..... 149
MINimum.....233	I2C
MODE.....233	ADDRess
POSITION	DIRection 140
MAXimum..... 234	MODE..... 139
MINimum 234	TYPe..... 139
SAVE	VALue..... 140
Auto.....235	CONDition 138
MAXimum.....234	DATA
MINimum235	SIZE 141
TOLerance235	VALue..... 141
TIMEbase	LIN
EXPand117	CONDition 152
MODE.....118	DATA
POSITION.....117	QUALifier 153
SCALE118	SIZE 154
WINDow	VALue..... 154
POSITION 118	ERRTYPE..... 155
SCALE..... 119	Identifier
TRIGger	VALue..... 156
ALTErnate.....135	SPI
BUS	CONDition 145
B1	DATA
CAN	MISO
	VALue 146
	MOSI
	VALue 147
	SIZE 146
	UART
	CONDition 142

RX	NREJ	123
DATA	PULSe	
SIze	TIME	134
VALue	WHEn	133
TX	PULSEWidth	
DATA	POLarity	128
SIze	RISEFall	
VALue	SLOP	130
THReshold	TIME	131
CH	WHEn	130
TYPe	RUNT	
COUPLE	POLarity	129
DELaY	TIME	130
EVENT	WHEn	129
LEVel	SOURce	123
SLOP	STATe	135
TIME	TIMEOut	
TYPe	TIMER	135
EDGE	WHEn	134
SLOP	TYPe	122
EXTERnal	VIDeo	
PRObe	FIELD	132
RATio	LINE	132
TYPe	POLarity	133
FREQuency	TYPe	131
HLEVel	USB	
HOLDoff	function check	4
LEVel	remote control interface	3
LLEVel		
MODE		