

Digital Storage Oscilloscope

GDS-2000E Series

PROGRAMMING MANUAL

GW INSTEK PART NO. Version 1.03, June 2015



ISO-9001 CERTIFIED MANUFACTURER

GW INSTEK

June 2015

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	5
Interface Configuration	5
COMMAND OVERVIEW	15
Command Syntax	15
List of Commands in Functional Order	17
COMMAND DETAILS	30
Common Commands	32
Acquisition Commands	38
Autoscale Commands	45
Vertical Commands	46
Math Commands	52
Cursor Commands	61
Display Commands	71
Hardcopy Commands	75
Measure Commands	78
Measurement Commands	103
Reference Commands	110
Run Command	113
Stop Command	113
Single Command	113
Force Command	114
Timebase Commands	115
Trigger Commands	118
System Commands	155
Save/Recall Commands	156
Ethernet Commands	161
Time Commands	162
Bus Decode Commands	163
Mark Commands	177

Search Commands	179
Label Commands	211
Segment Commands	217
DVM Commands	226
Go_NoGo Commands	228
Data Logging Commands	234
Remote Disk Commands	237
APPENDX.....	240
Error messages	240
INDEX.....	244

INTERFACE OVERVIEW

This manual describes how to use the GDS-2000E's remote command functionality and lists the command details. The Overview chapter describes how to configure the GDS-2000E USB and Ethernet remote control interface.

Interface Configuration

Configure USB Interface

USB Configuration	PC side connector	Type A, host
	GDS-2000E 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 GDS-2000E Download section. The driver automatically sets the GDS-2000E as a serial COM port (Shown as VPO in the PORTS node).

USB Functionality Check

Terminal Application

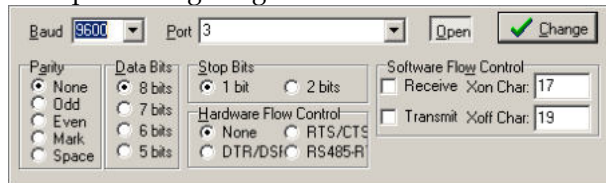
Invoke a terminal application such as RealTerm.

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?`





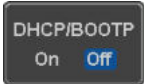
This should return the Manufacturer, Model number, Serial number, and Firmware version in the following format.

GW,GDS-2202E,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
		HTTP Port 80 (fixed)

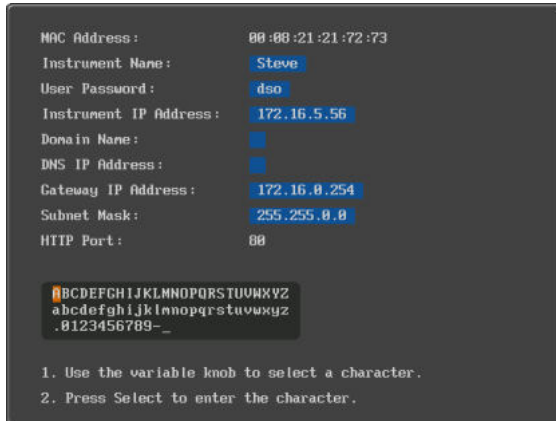
Background The Ethernet interface is used for remote control using a socket server connection. For details, please see the Socket Server section on page 9.

- 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. 

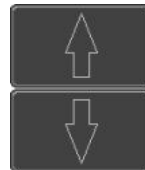


Note

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



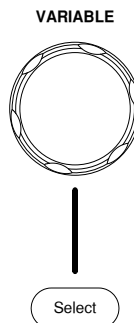
6. Use the *Up* and *Down* arrows on the side menu to navigate to each Ethernet configuration item.



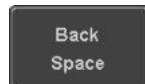
Items MAC Address, Instrument Name, User Password, Instrument IP Address, Domain Name, DNS IP Address, Gateway IP Address, Subnet Mask

Note: HTTP Port is fixed at 80.

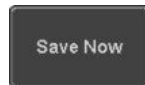
7. Use the *Variable* knob to highlight a character and use the *Select* key to choose a character.



Press *Backspace* to delete a character.



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



Configure Socket Server

The GDS-2000E 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 GDS-2000E. Page 7

2. Press the *Utility* key.



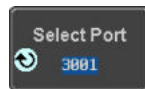
3. Press *I/O* from the bottom menu.



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

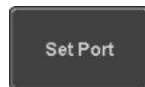


5. Press *Select Port* and choose the port number with the Variable knob.



Range 1024~65535

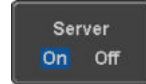
6. Press *Set Port* to confirm the port number.



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 | <ol style="list-style-type: none"> 1. Configure the IP address for the GDS-2000E. Page 7 2. Configure the socket port. Page 9 3. Start the NI Measurement and Automation Explorer (MAX) program. Using Windows, press: |
|-----------|---|



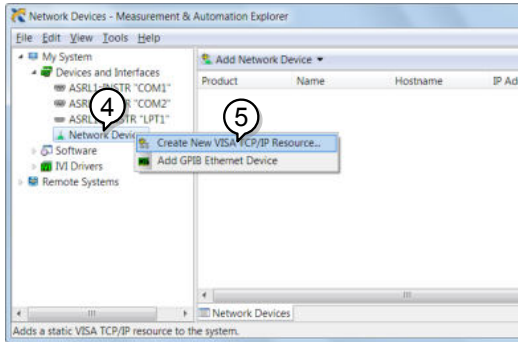
Start>All Programs>National Instruments>Measurement & Automation



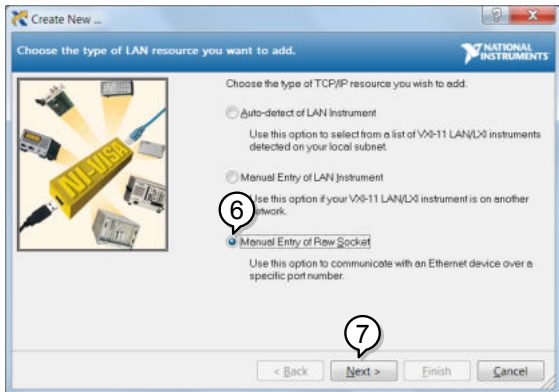
4. From the Configuration panel access;

My System>Devices and Interfaces>Network Devices

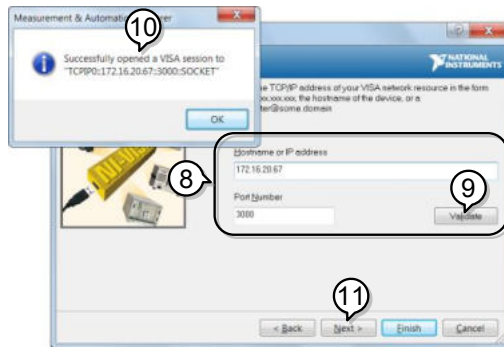
5. Right click *Network Devices* and select *Create New Visa TCP/IP Resource...*



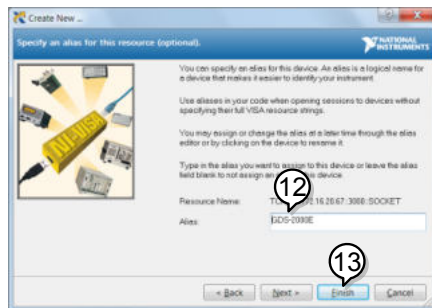
6. Select *Manual Entry of Raw Socket* from the popup window.
7. Click *Next*.



8. Enter the GDS-2000E'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*.



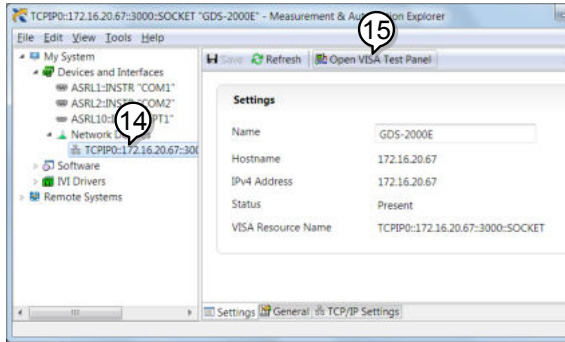
12. Choose an alias for the socket connection if you like.
13. Click *Finish* to finish the configuration.



14. The GDS-2000E will now appear under Network Devices in the Configuration Panel.

Functionality
Check

- Click the *Open Visa Test Panel* to send a remote command to the GDS-2000E.

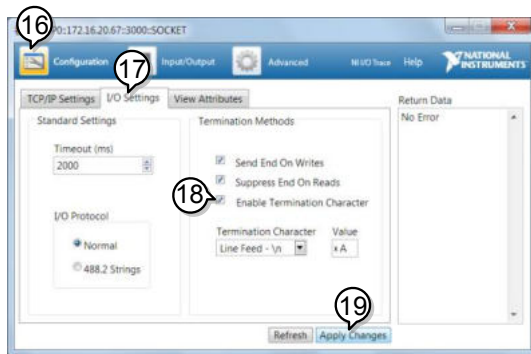


- Click on the *Configuration* icon.

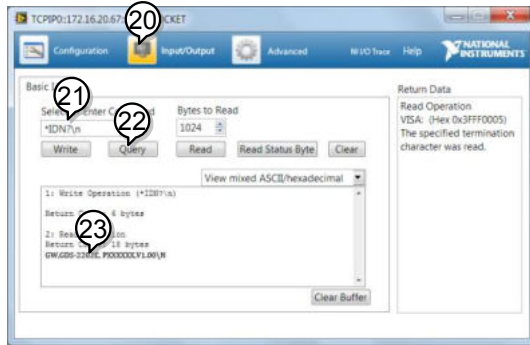
- Select the *I/O Settings* tab.

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

- 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,GDS-2202E,PXXXXXX,V1.00



COMMAND OVERVIEW

The Command overview chapter lists all GDS-2000E 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.

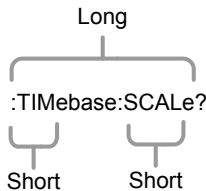
Command Syntax

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.



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
 2: single space
 3: parameter
 4: message terminator

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
<hr/>		
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
<hr/>		
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
Display	:DISPlay:INTensity:WAVEform	71
	:DISPlay:INTensity:GRATICule	72
	:DISPlay:INTensity:BACKLight	72
	:DISPlay:INTensity:BACKLight:AUTODim:ENABle ...	72
	DISPlay:INTENSITY:BACKLight:AUTODim:TIME	73
	:DISPlay:PERSistence	73
	:DISPlay:GRATICule	74
	:DISPlay:WAVEform	74
	:DISPlay:OUTPut	74

Hardcopy	:HARDcopy:START	75
	:HARDcopy:MODE	75
	:HARDcopy:PRINTINKSaver	76
	:HARDcopy:SAVEINKSaver	76
	:HARDcopy:SAVEFORMat	77
	:HARDcopy:ASSIGN.....	77
Measure	:MEASure:GATing.....	79
	:MEASure:SOURce	79
	:MEASure:METHod	80
	:MEASurement:REFLevel:PERCent:HIGH	80
	:MEASurement:REFLevel:PERCent:LOW	81
	:MEASurement:REFLevel:PERCent:MID	81
	:MEASurement:REFLevel:PERCent:MID2	81
	:MEASure:FALL.....	82
	:MEASure:FOVShoot.....	82
	:MEASure:FPreshoot	83
	:MEASure:FREQuency	83
	:MEASure:NWIDth	84
	:MEASure:PDUTy	84
	:MEASure:PERiod	85
	:MEASure:PWIDth	85
	:MEASure:RISe.....	86
	:MEASure:ROVShoot.....	86
	:MEASure:RPreshoot.....	87
	:MEASure:PPULSE.....	88
	:MEASure:NPULSE	88
	:MEASure:PEDGE	89
	:MEASure:NEDGE	89
	:MEASure:AMPlitude.....	90
	:MEASure:MEAN	90
	:MEASure:CMEan	91
	:MEASure:HIGH.....	92
	:MEASure:LOW.....	92
:MEASure:MAX.....	93	

	:MEASure:MIN	93
	:MEASure:PK2PK	94
	:MEASure:RMS	94
	:MEASure:CRMS	95
	:MEASure:AREa	96
	:MEASure:CARea	96
	:MEASure:FRRDelay	97
	:MEASure:FRFDelay.....	97
	:MEASure:FFRDelay.....	98
	:MEASure:FFFDelay.....	99
	:MEASure:LRRDelay	99
	:MEASure:LRFDelay.....	100
	:MEASure:LFRDelay.....	100
	:MEASure:LFFDelay.....	101
	:MEASure:PHAsE.....	102
<hr/>		
Measurement	:MEASUrement:MEAS<X>:SOURCE<X>.....	103
	:MEASUrement:MEAS<X>:TYPe.....	104
	:MEASUrement:MEAS<X>:STATE	104
	:MEASUrement:MEAS<X>:VALue.....	105
	:MEASUrement:MEAS<X>:MAXimum	106
	:MEASUrement:MEAS<X>:MEAN	107
	:MEASUrement:MEAS<X>:MINimum	107
	:MEASUrement:MEAS<X>:STDdev	108
	:MEASUrement:STATIstics:MODE	109
	:MEASUrement:STATIstics:WEIghting	109
	:MEASUrement:STATIstics	109
<hr/>		
Reference	:REF<X>:DISPlay.....	110
	:REF<X>:TIMebase:POSition	110
	:REF<X>:TIMebase:SCALE	111
	:REF<X>:OFFSet	111
	:REF<x>:SCALE	112
<hr/>		
Run	:RUN	113

Stop	:STOP	113
Single	:SINGle.....	113
Force	:FORCe.....	114
Timebase	:TIMebase:EXPand	115
	:TIMebase:POSition	115
	:TIMebase:SCALE	115
	:TIMebase:MODE.....	116
	:TIMebase:WINDow:POSition	116
	:TIMebase:WINDow:SCALE	117
Trigger	:TRIGger:FREQUency.....	120
	:TRIGger:TYPe	120
	:TRIGger:SOURce	121
	:TRIGger:COUPLE.....	121
	:TRIGger:NREJ	121
	:TRIGger:MODE.....	122
	:TRIGger:HOLDoff.....	122
	:TRIGger:LEVel.....	122
	:TRIGger:HLEVel.....	123
	:TRIGger:LLEVel.....	123
	:TRIGger:EDGE:SLOP	124
	:TRIGger:DElay:SLOP	124
	:TRIGger:DElay:TYPe.....	125
	:TRIGger:DElay:TIME.....	125
	:TRIGger:DElay:EVENT.....	125
	:TRIGger:DElay:LEVel	126
	:TRIGger:PULSEWidth:POLarity	126
	:TRIGger:RUNT:POLarity	127
	:TRIGger:RUNT:WHEn.....	127
	:TRIGger:RUNT:TIME.....	128
	:TRIGger:RISEFall:SLOP.....	128
	:TRIGger:RISEFall:WHEn	128

:TRIGger:RISEFall:TIME.....	129
:TRIGger:VIDeo:TYPe	129
:TRIGger:VIDeo:FIELD	130
:TRIGger:VIDeo:LINE.....	130
:TRIGger:VIDeo:POLarity	131
:TRIGger:PULSe:WHEn	131
:TRIGger:PULSe:TIME	132
:TRIGger:TIMEOut:WHEn.....	132
:TRIGger:TIMEOut:TIMER	133
:TRIGger:ALTErnate	133
:TRIGger:STATe	133
:TRIGger:EXTErnal:PROBe:TYPe.....	134
:TRIGger:EXTErnal:PROBe:RATio	134
:TRIGger:BUS:TYPe	135
:TRIGger:BUS:THReshold:CH<x>.....	135
:TRIGger:BUS:B1:I2C:CONDition	136
:TRIGger:BUS:B1:I2C:ADDReSS:MODE.....	136
:TRIGger:BUS:B1:I2C:ADDReSS:TYPe.....	137
:TRIGger:BUS:B1:I2C:ADDReSS:VALue	137
:TRIGger:BUS:B1:I2C:ADDReSS:DIRectioN.....	138
:TRIGger:BUS:B1:I2C:DATA:SIZE	139
:TRIGger:BUS:B1:I2C:DATA:VALue	139
:TRIGger:BUS:B1:UART:CONDition	140
:TRIGger:BUS:B1:UART:RX:DATA:SIZE	140
:TRIGger:BUS:B1:UART:RX:DATA:VALue	141
:TRIGger:BUS:B1:UART:TX:DATA:SIZE.....	142
:TRIGger:BUS:B1:UART:TX:DATA:VALue	142
:TRIGger:BUS:B1:SPI:CONDition	143
:TRIGger:BUS:B1:SPI:DATA:SIZE	143
:TRIGger:BUS:B1:SPI:DATA:MISO:VALue	144
:TRIGger:BUS:B1:SPI:DATA:MOSI:VALue	145
:TRIGger:BUS:B1:CAN:CONDition	145
:TRIGger:BUS:B1:CAN:FRAMeType	146
:TRIGger:BUS:B1:CAN:IDentifier:MODE.....	146
:TRIGger:BUS:B1:CAN:IDentifier:VALue	147

	:TRIGger:BUS:B1:CAN:IDentifier:DIRection	148
	:TRIGger:BUS:B1:CAN:DATA:QUALifier	148
	:TRIGger:BUS:B1:CAN:DATA:SIZE	149
	:TRIGger:BUS:B1:CAN:DATA:VALue	149
	:TRIGger:BUS:B1:LIN:CONDition	150
	:TRIGger:BUS:B1:LIN:DATA:QUALifier	151
	:TRIGger:BUS:B1:LIN:DATA:SIZE	152
	:TRIGger:BUS:B1:LIN:DATA:VALue	152
	:TRIGger:BUS:B1:LIN:ERRTYpe	153
	:TRIGger:BUS:B1:LIN:IDentifier:VALue	153
<hr/>		
System	:SYSTem:LOCK	155
commands	:SYSTem:ERRor	155
<hr/>		
Save/Recall	:RECALL:SETUp	156
	:RECALL:WAVEform	156
	:SAVE:IMAGe	157
	:SAVE:IMAGe:FILEFormat	157
	:SAVE:IMAGe:INKSaver	158
	:SAVE:SETUp	158
	:SAVE:WAVEform	159
	:SAVE:WAVEform:FILEFormat	160
<hr/>		
Ethernet	:ETHERnet:DHCP	161
Commands		
<hr/>		
Time Commands	:DATE	162
<hr/>		
Bus Decode	:BUS1	164
Commands	:BUS1:STATE	164
	:BUS1:TYPe	164
	:BUS1:INPut	165
	:BUS1:I2C:ADDReSS:RWINClude	165
	:BUS1:I2C:SCLK:SOURce	166
	:BUS1:I2C:SDA:SOURce	166
	:BUS1:UART:BITRate	166
	:BUS1:UART:DATABits	167

	:BUS1:UART:PARity	167
	:BUS1:UART:PACKet	168
	:BUS1:UART:EOFPAcket	168
	:BUS1:UART:TX:SOURce	168
	:BUS1:UART:RX:SOURce	169
	:BUS1:SPI:SCLK:POLARity	169
	:BUS1:SPI:SS:POLARity	169
	:BUS1:SPI:WORDSize	170
	:BUS1:SPI:BITORder	170
	:BUS1:SPI:SCLK:SOURce	170
	:BUS1:SPI:SS:SOURce.....	171
	:BUS1:SPI:MOSI:SOURce	171
	:BUS1:SPI:MISO:SOURce	171
	:BUS1:DISplay:FORMAt	172
	:LISTer:DATA	172
	:BUS1:CAN:SOURce.....	172
	:BUS1:CAN:PROBe.....	173
	:BUS1:CAN:SAMPLEpoint.....	173
	:BUS1:CAN:BITRate.....	173
	:BUS1:LIN:BITRate	174
	:BUS1:LIN:IDFORmat	174
	:BUS1:LIN:POLARity	175
	:BUS1:LIN:SAMPLEpoint	175
	:BUS1:LIN:SOURce	175
	:BUS1:LIN:STANDard	176
<hr/>		
Mark Commands	:MARK	177
	:MARK:CREATE.....	177
	:MARK:DELEte	178
<hr/>		
Search	:SEARCH:COPY	180
Commands	:SEARCH:STATE	181
	:SEARCH:TOTAL.....	181
	:SEARCH:TRIGger:TYPe	181
	:SEARCH:TRIGger:SOURce.....	182

:SEARCH:TRIGger:EDGE:SLOP	182
:SEARCH:TRIGger:LEVel	182
:SEARCH:TRIGger:HLEVel	183
:SEARCH:TRIGger:LLEVel	183
:SEARCH:TRIGger:PULSEWidth:POLarity	184
:SEARCH:TRIGger:RUNT:POLarity	184
:SEARCH:TRIGger:RISEFall:SLOP	185
:SEARCH:TRIGger:PULSE:WHEn	185
:SEARCH:TRIGger:PULSE:TIME	186
:SEARCH:TRIGger:RUNT:WHEn	186
:SEARCH:TRIGger:RUNT:TIME	187
:SEARCH:TRIGger:RISEFall:WHEn	187
:SEARCH:TRIGger:RISEFall:TIME	188
:SEARCH:TRIGger:BUS:TYPE	188
:SEARCH:TRIGger:BUS:B1:I2C:CONDition	189
:SEARCH:TRIGger:BUS:B1:I2C:ADDRes:MODE	189
:SEARCH:TRIGger:BUS:B1:I2C:ADDRes:TYPE	190
:SEARCH:TRIGger:BUS:B1:I2C:ADDRes:VALue	191
:SEARCH:TRIGger:BUS:B1:I2C:ADDRes :DIRection	191
:SEARCH:TRIGger:BUS:B1:I2C:DATA:SIZE	192
:SEARCH:TRIGger:BUS:B1:I2C:DATA:VALue	192
:SEARCH:TRIGger:BUS:B1:UART:CONDition	193
:SEARCH:TRIGger:BUS:B1:UART:RX:DATA:SIZE	194
:SEARCH:TRIGger:BUS:B1:UART:RX:DATA:VALue ..	195
:SEARCH:TRIGger:BUS:B1:UART:TX:DATA:SIZE	195
:SEARCH:TRIGger:BUS:B1:UART:TX:DATA:VALue ..	196
:SEARCH:TRIGger:BUS:B1:SPI:CONDition	197
:SEARCH:TRIGger:BUS:B1:SPI:DATA:SIZE	197
:SEARCH:TRIGger:BUS:B1:SPI:DATA:MISO:VALue ..	198
:SEARCH:TRIGger:BUS:B1:SPI:DATA:MOSI:VALue ..	199
:SEARCH:TRIGger:BUS:B1:CAN:CONDition	199
:SEARCH:TRIGger:BUS:B1:CAN:FRAMeType	200
SEARCH:TRIGger:BUS:B1:CAN:IDentifier:MODE ...	201
:SEARCH:TRIGger:BUS:B1:CAN:IDentifier:VALue ...	201

	:SEARCH:TRIGger:BUS:B1:CAN:IDentifier	
	:DIRection	202
	:SEARCH:TRIGger:BUS:B1:CAN:DATA:QUALifier.....	203
	:SEARCH:TRIGger:BUS:B1:CAN:DATA:SIZE	204
	:SEARCH:TRIGger:BUS:B1:CAN:DATA:VALue	204
	:SEARCH:TRIGger:BUS:B1:LIN:CONDition	205
	:SEARCH:TRIGger:BUS:B1:LIN:DATA:QUALifier	206
	:SEARCH:TRIGger:BUS:B1:LIN:DATA:SIZE	207
	:SEARCH:TRIGger:BUS:B1:LIN:DATA:VALue	207
	:SEARCH:TRIGger:BUS:B1:LIN:ERRTYPE	208
	:SEARCH:TRIGger:BUS:B1:LIN:IDentifier:VALue.....	208
	:SEARCH:FFTPeak:METHod	209
	:SEARCH:FFTPeak:METHod:MPEak.....	209
	:SEARCH:FFTPeak:SINfo.....	210
<hr/>		
Label Commands	:CHANnel<X>:LABel.....	211
	:CHANnel<X>:LABel:DISPlay	212
	:REF<X>:LABel	212
	:REF<X>:LABel:DISPlay	213
	:BUS1:LABel	214
	:BUS1:LABel:DISPlay	214
	:SET<X>:LABel	215
<hr/>		
Segment	:SEGMENTS:STATE	217
Commands	:SEGMENTS:CURRent	218
	:SEGMENTS:TOTALnum	218
	:SEGMENTS:TIME	219
	:SEGMENTS:DISPALL	219
	:SEGMENTS:MEASURE:MODE	219
	:SEGMENTS:MEASURE:PLOT:SOURCE	220
	:SEGMENTS:MEASURE:PLOT:DIVide	220
	:SEGMENTS:MEASURE:PLOT:SElect	221
	:SEGMENTS:MEASURE:PLOT:RESults.....	221
	:SEGMENTS:MEASURE:TABLE:SOURCE	222
	:SEGMENTS:MEASURE:TABLE:SElect.....	222

	:SEGMENTS:MEASURE:TABLE:LIST	223
	:SEGMENTS:MEASURE:TABLE:SAVE	223
	:SEGMENTS:SAVE	223
	:SEGMENTS:SAVE:SOURCE	224
	:SEGMENTS:SAVE:SELECT:START	224
	:SEGMENTS:SAVE:SELECT:END	225
<hr/>		
DVM Commands	:DVM:STATE	226
	:DVM:SOURCE	226
	:DVM:MODE	227
	:DVM:VALUE	227
<hr/>		
Go-NoGo	:GONOGO:CLEAR	228
Commands	:GONOGO:EXECUTE	228
	:GONOGO:FUNCTION	229
	:GONOGO:NGCOUNT	229
	:GONOGO:NGDEFINE	229
	:GONOGO:SOURCE	230
	:GONOGO:VIOLATION	230
	:GONOGO:SCRIPT	230
	:TEMPLATE:MODE	231
	:TEMPLATE:MAXIMUM	231
	:TEMPLATE:MINIMUM	231
	:TEMPLATE:POSITION:MAXIMUM	232
	:TEMPLATE:POSITION:MINIMUM	232
	:TEMPLATE:SAVE:MAXIMUM	232
	:TEMPLATE:SAVE:MINIMUM	233
	:TEMPLATE:TOLERANCE	233
	:TEMPLATE:SAVE:AUTO	233
<hr/>		
Data Logging	:DATALOG:STATE	234
Commands	:DATALOG:SOURCE	234
	:DATALOG:SAVE	235
	:DATALOG:INTERVAL	235
	:DATALOG:DURATION	236

Remote Disk	:REMOTEDisk:IPADdress	237
Commands	:REMOTEDisk:PATHName	237
	:REMOTEDisk:USERName	237
	:REMOTEDisk:PASSWord	238
	:REMOTEDisk:MOUNT.....	238
	:REMOTEDisk:AUTOMount.....	239

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 page17.

Common Commands	32
Acquisition Commands	38
Autoscale Commands	45
Vertical Commands	46
Math Commands	52
Cursor Commands	61
Display Commands	71
Hardcopy Commands.....	75
Measure Commands	78
Measurement Commands	103
Reference Commands	110
Run Command	113
Stop Command	113
Single Command.....	113
Force Command.....	114
Timebase Commands.....	115
Trigger Commands.....	118
System Commands	155
Save/Recall Commands.....	156
Ethernet Commands	161
Time Commands.....	162
Bus Decode Commands	163
Mark Commands.....	177
Search Commands	179

Label Commands	211
Segment Commands	217
DVM Commands	226
Go_NoGo Commands	228
Data Logging Commands	234
Remote Disk Commands	237

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,GDS-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 GDS-2000E (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

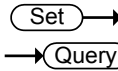
Bit Weight	Bit#	Weight	Event	Description
	0	1		Not used
	1	2		Not used
	2	4		Not used
	3	8		Not used
	4	16	MAV	Message Available Bit
	5	32	ESB	Event Status Bit
	6	64	MSS	Master Summary Bit
	6	64	RQS	Request Service Bit
	7	128		Not used

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%; background-color: #e0e0e0;">SAMPlE</td> <td>Sample mode sampling</td> </tr> <tr> <td style="background-color: #e0e0e0;">PDETECT</td> <td>Peak detect sampling</td> </tr> <tr> <td style="background-color: #e0e0e0;">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.						

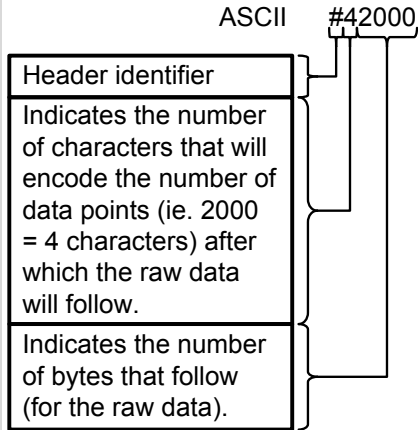
→ Query

: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%; background-color: #e0e0e0;"><string></td> <td>Returns acquisition settings followed by raw waveform block data.</td> </tr> <tr> <td style="background-color: #e0e0e0;"><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.....

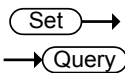
The actual value of a data point can be calculated with the following formula: (Decimal value of hex data/AD Factor) * vertical scale.

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.....



: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>	

:ACquire:INTERpolation Set → → Query

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

:ACquire:RECOrdlength Set → → 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
 (Set) →
 → (Query)

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

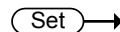
:AUTOSet	45
:AUTORSET:MODE	45

:AUTOSet

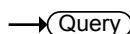


Description Runs the Autoset function to automatically configure the horizontal scale, vertical scale, and trigger according to the input signal.

Syntax :AUTOSet



:AUTORSET:MODE



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?
FITSREEN

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 GDS-2000E 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

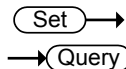
Description	Sets or returns the vertical scale. The scale depends on the probe attenuation factor. Note the probe attenuation factor should be set before the scale.	
-------------	---	--

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

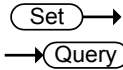
Return parameter Returns the source for the source 1 or 2.

Example :MATH:DUAL:SOURce1 CH1
Sets source1 as channel 1.

:MATH:DUAL:OPERator
 →
 →

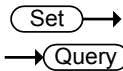
Description Sets the math operator for dual math operations.

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	Channel 1 to 4
	REF1~4	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	Linear units (Vrms)
	DB	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:HORizontal: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

MATHVAR:VAR<X>

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

:MATH:ADVanced:POSition

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

:MATH:ADVanced:SCALE

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


Set →
 → Query

: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.	

Set →
 → Query

:CURSor:SOURce		
Description	Sets or queries the cursor source.	
Syntax	:CURSor:SOURce {CH1 CH2 CH3 CH4 REF1 REF2 REF3 REF4 MATH BUS1 ? }	
Parameter	CH1~CH4	Channel 1 to 4
	REF1~4	Reference waveform 1 to 4
	MATH	Math 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.	

:CURSor:V1Position

Set →

→ **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 →

→ **Query**

:CURSor:V2Position

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.....	71
:DISPlay:INTensity:GRATICule.....	72
:DISPlay:INTensity:BACKLight	72
:DISPlay:INTensity:BACKLight:AUTODim:ENABLE ...	72
DISPlay:INTENSITY:BACKLight:AUTODim:TIME	73
:DISPlay:PERSiStence	73
:DISPlay:GRATICule	74
:DISPlay:WAVEform.....	74
:DISPlay:OUTPut	74

: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%.

Set →

→ Query

:DISPlay:INTensity:GRATicule

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%.

Set →

→ Query

: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%.

Set →

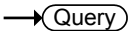
→ Query

**:DISPlay:INTensity:BACKLight:AUTODim
:ENABLE**

Description	Sets or queries the display auto-dim function.
Syntax	:DISPlay:INTensity:BACKLight:AUTODim:ENABLE {OFF ON ?}

Parameter/	OFF	Turn auto-dim on.
Return parameter	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/	<NR1>	1 ~ 180 minutes. Time in minutes.
Return parameter		

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

:DISPlay:PERSistence 


Description Sets or queries the waveform persistence level.

Syntax :DISPlay:PERSistence { INFIinite | 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
	INFIinite	Infinite persistence
	OFF	No persistence

Return Parameter	<NR3>	Returns the persistence time.
	INFIinite	Infinite persistence
	OFF	No persistence


Example :DISPlay:PERSistence 2.0E+0
Sets the persistence to 2 seconds.

:DISPlay:GRATicule




Description	Sets or queries graticule display type.			
Syntax	:DISPlay:GRATicule {FULL GRID CROSSs FRAMe ?}			
Parameter	FULL		CROSSs	
	FRAMe		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.

Hardcopy Commands

:HARDcopy:START	75
:HARDcopy:MODE	75
:HARDcopy:PRINTINKSaver	76
:HARDcopy:SAVEINKSaver	76
:HARDcopy:SAVEFORMat	77
:HARDcopy:ASSIGN	77

:HARDcopy:START

Set →

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

Set →
→ **Query**

Description	Sets or queries whether hardcopy is set to print or save.
Syntax	:HARDcopy:MODE { PRINT SAVE ? }

Related Commands :HARDcopy:START

Parameter	PRINT	Print mode
	SAVE	Save mode

Return parameter Returns the mode.(PRINT/SAVE)

Example :HARDcopy:MODE PRINT
Sets hardcopy to print.

Set →

→ Query

:HARDcopy:PRINTINKSaver

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.

:HARDcopy:SAVEFORMat



Description	Sets or queries the image save file type.				
Syntax	:HARDcopy:SAVEFORMat { PNG BMP ? }				
Related Commands	:HARDcopy:START :HARDcopy:MODE				
Parameter	<table border="1"> <tr> <td>PNG</td> <td>PNG file format</td> </tr> <tr> <td>BMP</td> <td>BMP file format</td> </tr> </table>	PNG	PNG file format	BMP	BMP file format
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.				


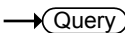
:HARDcopy:ASSIGN



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	<table border="1"> <tr> <td>IMAGe</td> <td>Save image files.</td> </tr> <tr> <td>WAVEform</td> <td>Save waveforms.</td> </tr> <tr> <td>SETUp</td> <td>Save the panel setup.</td> </tr> <tr> <td>ALL</td> <td>Save All (image, waveform,setup)</td> </tr> </table>	IMAGe	Save image files.	WAVEform	Save waveforms.	SETUp	Save the panel setup.	ALL	Save All (image, waveform,setup)
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.....	79
:MEASure:SOURce	79
:MEASure:METHOD	80
:MEASUrement:REFLevel:PERCent:HIGH	80
:MEASUrement:REFLevel:PERCent:LOW	81
:MEASUrement:REFLevel:PERCent:MID	81
:MEASUrement:REFLevel:PERCent:MID2	81
:MEASure:FALL.....	82
:MEASure:FOVShoot.....	82
:MEASure:FPReshoot	83
:MEASure:FREQuency	83
:MEASure:NWIDth	84
:MEASure:PDUTy	84
:MEASure:PERiod	85
:MEASure:PWIDth	85
:MEASure:RISe.....	86
:MEASure:ROVShoot.....	86
:MEASure:RPReshoot.....	87
:MEASure:PPULSE.....	88
:MEASure:NPULSE	88
:MEASure:PEDGE	89
:MEASure:NEDGE	89
:MEASure:AMPlitude.....	90
:MEASure:MEAN	90
:MEASure:CMEan	91
:MEASure:HIGH	92
:MEASure:LOW.....	92
:MEASure:MAX.....	93
:MEASure:MIN	93
:MEASure:PK2PK.....	94
:MEASure:RMS	94
:MEASure:CRMS.....	95

:MEASure:AREa	96
:MEASure:CARea	96
:MEASure:FRRDelay	97
:MEASure:FRFDelay.....	97
:MEASure:FFRDelay.....	98
:MEASure:FFFDelay.....	99
:MEASure:LRRDelay	99
:MEASure:LRFDelay.....	100
:MEASure:LFRDelay.....	100
:MEASure:LFFDelay.....	101
:MEASure:PHAsE.....	102

: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.

Set →

→ Query

:MEASure:METHOD

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.

Set →

→ Query

:MEASUrement:REFLevel:PERCent:HIGH

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
 →
 → 

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
 →
 → 

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
 →
 → 

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	:MEASure:SOURce1 CH1 :MEASure:FALL? Selects Channel 1 as the source, and then measures the fall time.	

:MEASure:FOVShoot

→ **Query**

Description	Returns the fall overshoot amplitude.	
Syntax	:MEASure:FOVShoot{?}	
Related Commands	:MEASure:SOURce<X>	
Return parameter	<NR3> Chan Off	Returns the fall overshoot as a percentage 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	<pre>:MEASure:SOURce1 CH1 :MEASure:PDUTy? 5.000E+01</pre> <p>Selects Channel 1, and then measures the positive duty cycle.</p>	

: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	<pre>:MEASure:SOURce1 CH1 :MEASure:PERiod? 1.0E-3</pre> <p>Selects Channel 1, and then measures the period.</p>	

: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	<pre>:MEASure:SOURce1 CH1 :MEASure:ROVShoot? 5.00E+00</pre> 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	<pre>:MEASure:SOURce1 CH1 :MEASure:RPReshoot? 2.13E-2</pre> 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	:MEASure:SOURce1 CH1	
	:MEASure:PPULSE?	
	6.000E+00	
	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	<pre>:MEASure:SOURce1 CH1 :MEASure:NEDGE? 1.100E+01</pre> <p>Selects Channel 1, and then measures the number of negative edges.</p>	

: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	<pre>:MEASure:SOURce1 CH1 :MEASure:AMplitude? 3.76E-3</pre> <p>Selects Channel 1, and then measures the amplitude.</p>	

: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>	Returns the mean.
	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: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>	Returns the cyclic mean.
	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:CMEan? 9.480E-01 Selects Channel 1, and then measures the mean value of the first period.	

:MEASure:HIGH

→ Query

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	:MEASure:SOURce1 CH1	
	:MEASure:HIGH?	
	3.68E-3	
	Selects Channel 1, and then measures the high voltage/current.	

:MEASure:LOW

→ Query

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 :MEASure:SOURce1 CH1
 :MEASure:SOURce2 CH2
 :MEASure:FRFDelay?
 3.43E-6
 Select channel 1 and 2 as source1/2, and then measures FRF.

:MEASure:FFRDelay → Query

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 :MEASure:SOURce1 CH1
 :MEASure:SOURce2 CH2
 :MEASure:FRRDelay?
 -8.56E-6
 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: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>	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:LRFDelay?
 -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>	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: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: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>.....	103
:MEASUrement:MEAS<X>:TYPE.....	104
:MEASUrement:MEAS<X>:STATE	104
:MEASUrement:MEAS<X>:VALue.....	105
:MEASUrement:MEAS<X>:MAXimum	106
:MEASUrement:MEAS<X>:MEAN	107
:MEASUrement:MEAS<X>:MINimum	107
:MEASUrement:MEAS<X>:STDdev	108
:MEASUrement:STATIstics:MODE	109
:MEASUrement:STATIstics:WEIghting	109
:MEASUrement:STATIstics	109

: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 | ?}

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.
	<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:MINImum?
 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.	

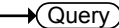
:MEASUREMENT:STATISTICS (Set) →

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

Reference Commands

:REF<X>:DISPlay	110
:REF<X>:TIMebase:POSition	110
:REF<X>:TIMebase:SCALe	111
:REF<X>:OFFSet.....	111
:REF<x>:SCALe	112

: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.

Set →
 → Query

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



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

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

Timebase Commands

:TIMebase:EXPand	115
:TIMebase:POSition	115
:TIMebase:SCALe.....	115
:TIMebase:MODE	116
:TIMebase:WINDow:POSition	116
:TIMebase:WINDow:SCALe	117

:TIMebase:EXPand (Set) → → (Query)

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.

(Set) →

:TIMebase:POSition → (Query)

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) →

:TIMebase:SCALe → (Query)

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 →

:TIMebase:MODE

→ Query

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 →

:TIMebase:WINDow:POSition

→ Query

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 

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.....	120
:TRIGger:TYPe	120
:TRIGger:SOURce.....	121
:TRIGger:COUPle.....	121
:TRIGger:NREJ	121
:TRIGger:MODe.....	122
:TRIGger:HOLDoff.....	122
:TRIGger:LEVel.....	122
:TRIGger:HLEVel.....	123
:TRIGger:LLEVel.....	123
:TRIGger:EDGE:SLOP.....	124
:TRIGger:DELaY:SLOP	124
:TRIGger:DELaY:TYPe.....	125
:TRIGger:DELaY:TIME.....	125
:TRIGger:DELaY:EVENT.....	125
:TRIGger:DELaY:LEVel	126
:TRIGger:PULSEWidth:POLarity	126
:TRIGger:RUNT:POLarity	127
:TRIGger:RUNT:WHEn.....	127
:TRIGger:RUNT:TIME.....	128
:TRIGger:RISEFall:SLOP.....	128
:TRIGger:RISEFall:WHEn	128
:TRIGger:RISEFall:TIME	129
:TRIGger:VIDeo:TYPe	129
:TRIGger:VIDeo:FIELD	130
:TRIGger:VIDeo:LINE	130
:TRIGger:VIDeo:POLarity	131
:TRIGger:PULSE:WHEn	131
:TRIGger:PULSE:TIME	132
:TRIGger:TIMEOut:WHEn.....	132
:TRIGger:TIMEOut:TIMER	133
:TRIGger:ALTErnate	133

:TRIGger:STATe	133
:TRIGger:EXTERnal:PROBe:TYPe.....	134
:TRIGger:EXTERnal:PROBe:RATio	134
:TRIGger:BUS:TYPe	135
:TRIGger:BUS:THReshold:CH<x>.....	135
:TRIGger:BUS:B1:I2C:CONDition	136
:TRIGger:BUS:B1:I2C:ADDRes:MODE.....	136
:TRIGger:BUS:B1:I2C:ADDRes:TYPe.....	137
:TRIGger:BUS:B1:I2C:ADDRes:VALue	137
:TRIGger:BUS:B1:I2C:ADDRes:DIRection.....	138
:TRIGger:BUS:B1:I2C:DATA:SIze	139
:TRIGger:BUS:B1:I2C:DATA:VALue.....	139
:TRIGger:BUS:B1:UART:CONDition	140
:TRIGger:BUS:B1:UART:RX:DATA:SIze	140
:TRIGger:BUS:B1:UART:RX:DATA:VALue	141
:TRIGger:BUS:B1:UART:TX:DATA:SIze.....	142
:TRIGger:BUS:B1:UART:TX:DATA:VALue	142
:TRIGger:BUS:B1:SPI:CONDition	143
:TRIGger:BUS:B1:SPI:DATA:SIze	143
:TRIGger:BUS:B1:SPI:DATA:MISO:VALue	144
:TRIGger:BUS:B1:SPI:DATA:MOSI:VALue	145
:TRIGger:BUS:B1:CAN:CONDition.....	145
:TRIGger:BUS:B1:CAN:FRAMeType	146
:TRIGger:BUS:B1:CAN:IDentifier:MODE.....	146
:TRIGger:BUS:B1:CAN:IDentifier:VALue	147
:TRIGger:BUS:B1:CAN:IDentifier:DIRection.....	148
:TRIGger:BUS:B1:CAN:DATA:QUALifier.....	148
:TRIGger:BUS:B1:CAN:DATA:SIze.....	149
:TRIGger:BUS:B1:CAN:DATA:VALue.....	149
:TRIGger:BUS:B1:LIN:CONDition	150
:TRIGger:BUS:B1:LIN:DATA:QUALifier.....	151
:TRIGger:BUS:B1:LIN:DATA:SIze	152
:TRIGger:BUS:B1:LIN:DATA:VALue	152
:TRIGger:BUS:B1:LIN:ERRTYPe.....	153
:TRIGger:BUS:B1:LIN:IDentifier:VALue.....	153

: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 BUS TIMEOut ? }
Parameter	EDGE Edge trigger DELay Delay trigger PULSEWidth Pulse width trigger VIDEo Video trigger RUNT Runt trigger RISEFall Rise and fall trigger BUS Bus trigger TIMEOut Timeout trigger
Return parameter	Returns the trigger type.
Example	:TRIGger:TYPe EDGE Sets the trigger type to edge.

Set →
 → Query

: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.	

Set →
 → Query

: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.	

Set →
 → Query

: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:NRE| 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.

(Set) →

→ (Query)

: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.

(Set) →

→ (Query)

: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 →

:TRIGger:DElay:TYPE

→ Query

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 →

:TRIGger:DElay:TIME

→ Query

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 →

:TRIGger:DElay:EVENT

→ Query

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 (Set) →
→ (Query)

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 (Set) →
→ (Query)

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 (Set) →
→ (Query)

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

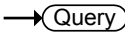
Description Sets or queries the video trigger line.

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 (Set) →
 → (Query)

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 (Set) →
 → (Query)

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.

Set →
 → Query

: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

Set →
 → Query

: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.

→ Query

: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	<code>:TRIGger:EXTERnAl:PROBE:TYPE { VOLTage CURRent ? }</code>	
Related commands	<code>:TRIGger:EXTERnAl:PROBE:RATio</code>	
Parameter	VOLTage	Voltage
	CURRent	Current

Return parameter Returns the probe type.

Example `:TRIGger:EXTERnAl:PROBE:TYPE?`
 `CURRENT`

Set →
 → Query

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 → Query

Description	Returns the current bus type.	
Syntax	:TRIGger:BUS:TYPe?	
Return parameter	I2C	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> Set →
→ Query

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 (Set) →
→ (Query)

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 I²C bus trigger condition.

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

:TRIGger:BUS:B1:I2C:ADDRess:MODE (Set) →
→ (Query)

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 currently set to 7 bits.

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

Description Sets the I²C 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:ADDRess:TYPE?
CBUS

:TRIGger:BUS:B1:I2C:ADDRess:VALue  →
→ 

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:ADDRess:VALue {<string> | ? }

Related commands :TRIGger:BUS:B1:I2C:ADDRess: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:ADDRess:MODE ADDR7
 :TRIGger:BUS:B1:I2C:ADDRess:VALue "xxx0101"
 Sets the address to XXX0101

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

:TRIGger:BUS:B1:I2C:ADDRess:DIRection 

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:ADDRess: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 (Set) →
→ (Query)

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.	

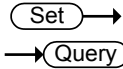
:TRIGger:BUS:B1:I2C:DATA:VALue (Set) →
→ (Query)

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	<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: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.

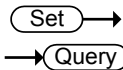
Syntax :TRIGger:BUS:B1:UART:CONDition { RXSTART | RXDATA | RXENDPacket | TXSTART | TXDATA | TXENDPacket | TXPARityerr | RXPARTyerr | ? }

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.
	RXPARTyerr	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.
	TXPARTyerr	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



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.	


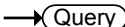
:TRIGger:BUS:B1:UART:RX:DATA:VALue 


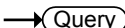
Description	Sets or queries the triggering data value for the UART bus when the bus is set to trigger on Rx Data.	
Syntax	:TRIGger:BUS:B1:UART:RX:DATA:VALue {<string> ? }	
Related commands	:TRIGger:BUS:B1:UART:RX:DATA:SIZE	
Parameter	<sting>	The number of characters in the string depends on the data 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.

Example1 :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	<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: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  

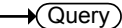
Description Sets or queries the SPI triggering condition.

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.

: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

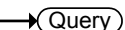
Parameter	<string>	The number of characters in the string depends on the data size setting. The string must be enclosed in double quotes, "string". x = don't care 1 = binary 1 0 = binary 0
-----------	----------	--

Return Parameter Returns the data value.

Example1 :TRIGger:BUS:B1: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

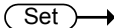


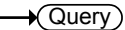


:TRIGger:BUS:B1:SPI:DATA:MOSI:VALue

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	<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 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

Description	Sets or returns the CAN trigger condition.	
Syntax	:TRIGger:BUS:B1:CAN:CONDition {SOF FRAMeType IDentifier DATA IDANDDATA EOF ACKMISS STUFFERR ?}	
Parameter/	SOF	Triggers on a start of frame

Return parameter	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

Set →

→ Query

:TRIGger:BUS:B1:CAN:FRAMeType

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.

Set →

→ Query

: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

:TRIGger:BUS:B1:CAN:IDentifier:DIRection (Set) →
→ (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	

:TRIGger:BUS:B1:CAN:DATA:QUALifier (Set) →
→ (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

Set →

→ Query

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

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

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
```

Set →
→ Query

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
```

Set →
 → Query

:TRIGger:BUS:B1:LIN:DATA:SIZE	
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

Set →
 → Query

:TRIGger:BUS:B1:LIN:DATA:VALue	
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
```

Set →

→ Query

: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
```

Set →

→ Query

: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	155
:SYSTem:ERRor.....	155

: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.	

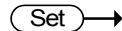
:SYSTem:ERRor  

Description	Queries the error queue. See the appendix on page 240 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	156
:RECALL:WAVEform	156
:SAVE:IMAGe	157
:SAVE:IMAGe:FILEFormat.....	157
:SAVE:IMAGe:INKSaver.....	158
:SAVE:SETUp	158
:SAVE:WAVEform.....	159
:SAVE:WAVEform:FILEFormat	160

: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



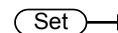
Description Saves a screen image to the assigned file path with a specified filename.

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

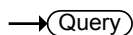
Related commands :SAVe:IMAGe:FILEFormat
 :SAVe:IMAGe:INKSaver

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.	

:SAVE:SETUp

Set →

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, CH1~CH4, Math, REF1~4
	<X> 1,2,3,4 (REF1, REF2, REF3, REF4)
	W1~W20 Wave1~Wave20
	ALL All the displayed waveforms on screen.
	File path 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 (Set) →
 ← (Query)

Description	Sets the waveform savefile format.	
Syntax	:SAVe:WAVEform:FILEFormat {LSF DCSV FCSV ?}	
Parameter	LSF	Sets the file format to the GDS 2000E'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 Commands

:ETHERnet:DHCP161

: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 Commands

:DATE162

:DATE



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	164
:BUS1:STATE	164
:BUS1:TYPe	164
:BUS1:INPut.....	165
:BUS1:I2C:ADDRes:RWINClude.....	165
:BUS1:I2C:SCLK:SOURce	166
:BUS1:I2C:SDA:SOURce.....	166
:BUS1:UART:BITRate.....	166
:BUS1:UART:DATABits	167
:BUS1:UART:PARItY.....	167
:BUS1:UART:PACKEt	168
:BUS1:UART:EOFPacket	168
:BUS1:UART:TX:SOURce	168
:BUS1:UART:RX:SOURce	169
:BUS1:SPI:SCLK:POLARity	169
:BUS1:SPI:SS:POLARity.....	169
:BUS1:SPI:WORDSize	170
:BUS1:SPI:BITORder	170
:BUS1:SPI:SCLK:SOURce	170
:BUS1:SPI:SS:SOURce.....	171
:BUS1:SPI:MOSI:SOURce	171
:BUS1:SPI:MISO:SOURce	171
:BUS1:DISPlay:FORMAt	172
:LISTer:DATA	172
:BUS1:CAN:SOURce.....	172
:BUS1:CAN:PROBe.....	173
:BUS1:CAN:SAMPLEpoint.....	173
:BUS1:CAN:BITRate.....	173
:BUS1:LIN:BITRate	174
:BUS1:LIN:IDFORmat	174
:BUS1:LIN:POLARity	175
:BUS1:LIN:SAMPLEpoint	175

:BUS1:LIN:SOURce175
 :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 →
→ Query

:BUS1:STATE

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 →
→ Query

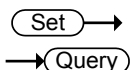
:BUS1:TYPE

Description	Sets or queries the type of bus.	
Syntax	:BUS1:TYPE { UART I2C SPI 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.
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



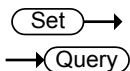
Description Sets or returns the analog source.

Syntax :BUS1:INPut {ANALog | ?}

Parameter/Return parameter	ANALog	Sets the source to the analog inputs.
----------------------------	--------	---------------------------------------

Example1 :BUS1:INPut?
>ANALOG

:BUS1:I2C:ADDRess:RWINClude



Description Sets or queries whether the read/write bit is included in the I²C address.

Syntax :BUS1:I2C:ADDRess: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:ADDRess:RWINClude ON
Includes the R/W bit in the I²C address.

:BUS1:I2C:SCLK:SOURce (Set) →
→ (Query)

Description	Sets or queries which channel is used for the I ² C 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.

:BUS1:I2C:SDA:SOURce (Set) →
→ (Query)

Description	Sets or queries which channel is used for the I ² C 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.

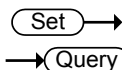
:BUS1:UART:BITRate (Set) →
→ (Query)

Description	Sets or queries the UART bit rate.			
Syntax	:BUS1:UART:BITRate { <NR1> ? }			
Parameter/Return parameter	<NR1>	UART bit rate (0~31)		
	<NR1>	Rate (bps)	<NR1>	Rate (bps)
	0	50	16	15200
	1	75	17	19200
	2	110	18	28800
	3	134	19	31250

4	150	20	38400
5	300	21	56000
6	600	22	57600
7	1200	23	76800
8	1800	24	115200
9	2000	25	128000
10	2400	26	230400
11	3600	27	460800
12	4800	28	921600
13	7200	29	1382400
14	9600	30	1843200
15	14400	31	2764800

Example :BUS1:UART:BITRate 10
Sets the bit rate to 2400.

:BUS1:UART:DATABits



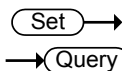
Description Sets or queries the number UART data for bus 1.

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.

:BUS1:UART:PARItY



Description Sets or queries the UART bus parity.

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

Syntax	:BUS1:UART:TX:SOURce { OFF CH1 CH2 CH3 CH4 ? }	
Parameter/Return parameter	OFF CH1 to CH4	Off, no Tx source Analog channels CH1 to CH4
Example	:BUS1:UART:TX:SOURce CH1 Sets channel 1 as the Tx 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 parameter	OFF CH1 to CH4	Off, no Rx source Analog channels CH1 to CH4
Example	:BUS1:UART:RX:SOURce CH1 Sets channel 1 as the Rx source.	

→
 →

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 RISE	Sets the polarity to falling edge. Sets the polarity to rising edge.
Example	:BUS1:SPI:SCLK:POLARity FALL Sets the polarity to falling edge.	

→
 →

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.	

Set →

→ Query

:BUS1:SPI:WORDSize

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.	

Set →

→ Query

: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.	

Set →

→ Query

: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 (Set) →
→ (Query)

Description Sets or queries which channel is used for the SPI SS source.

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

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.

:BUS1:SPI:MOSI:SOURce (Set) →
→ (Query)

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.

:BUS1:SPI:MISO:SOURce (Set) →
→ (Query)

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

parameter OFF No MISO source.

Example :BUS1:SPI:MISO:SOURce CH1
 Sets channel CH1 as the SPI MISO source.

(Set) →

:BUS1:DISplay:FORMAt →(Query)

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.

:BUS1:CAN:PROBe (Set) →
→ (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.

:BUS1:CAN:BITRate (Set) →
→ (Query)

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

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

Parameter/Return RATE10K 10 kbps

parameter	RATE20K	20 kbps
	RATE50K	50 kbps
	RATE125K	125 kbps
	RATE250K	250 kbps
	RATE500K	500 kbps
	RATE800K	800 kbps
	RATE1M	1 Mbps

Example :BUS1:CAN:BITRate?
 >RATE250K
 :BUS1:CAN:BITRate rate10k
 :BUS1:CAN:BITRate?
 >RATE10K

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>	1200, 2400, 4800, 9600, 10417, 19200

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

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.

:BUS1:LIN:POLARity (Set) →
→ (Query)

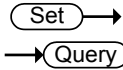
Description	Sets or returns the LIN polarity.	
Syntax	:BUS1:LIN:POLARity {NORMal INVerted ?}	
Parameter/Return parameter	NORMal	Normal LIN polarity
	INVerted	Inverted LIN polarity
Example	:BUS1:LIN:POLARity? NORMAL Returns the LIN polarity.	

: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.	

:BUS1:LIN:SOURce (Set) →
→ (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.	



:BUS1:LIN:STANDARD

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	<pre>:BUS1:LIN:STANDARD? >BOTH</pre> <p>Returns the LIN standard.</p>	

Mark Commands

:MARK	177
:MARK:CREATE.....	177
:MARK:DELEte	178

: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



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:COPIY	180
:SEARCH:STATE	181
:SEARCH:TOTAL.....	181
:SEARCH:TRIGger:TYPe	181
:SEARCH:TRIGger:SOURce.....	182
:SEARCH:TRIGger:EDGE:SLOP	182
:SEARCH:TRIGger:LEVel	182
:SEARCH:TRIGger:HLEVel	183
:SEARCH:TRIGger:LLEVel	183
:SEARCH:TRIGger:PULSEWidth:POLarity	184
:SEARCH:TRIGger:RUNT:POLarity	184
:SEARCH:TRIGger:RISEFall:SLOP.....	185
:SEARCH:TRIGger:PULSe:WHEn.....	185
:SEARCH:TRIGger:PULSe:TIme.....	186
:SEARCH:TRIGger:RUNT:WHEn	186
:SEARCH:TRIGger:RUNT:TIme.....	187
:SEARCH:TRIGger:RISEFall:WHEn	187
:SEARCH:TRIGger:RISEFall:TIme	188
:SEARCH:TRIGger:BUS:TYPe.....	188
:SEARCH:TRIGger:BUS:B1:I2C:CONDition.....	189
:SEARCH:TRIGger:BUS:B1:I2C:ADDRes:MODE	189
:SEARCH:TRIGger:BUS:B1:I2C:ADDRes:TYPe	190
:SEARCH:TRIGger:BUS:B1:I2C:ADDRes:VALue	191
:SEARCH:TRIGger:BUS:B1:I2C:ADDRes	
:DIRection	191
:SEARCH:TRIGger:BUS:B1:I2C:DATa:SIze.....	192
:SEARCH:TRIGger:BUS:B1:I2C:DATa:VALue.....	192
:SEARCH:TRIGger:BUS:B1:UART:CONDition.....	193
:SEARCH:TRIGger:BUS:B1:UART:RX:DATa:SIze	194
:SEARCH:TRIGger:BUS:B1:UART:RX:DATa:VALue ..	195
:SEARCH:TRIGger:BUS:B1:UART:TX:DATa:SIze	195
:SEARCH:TRIGger:BUS:B1:UART:TX:DATa:VALue ..	196
:SEARCH:TRIGger:BUS:B1:SPI:CONDition.....	197

:SEARCH:TRIGger:BUS:B1:SPI:DATA:SIZE.....197
 :SEARCH:TRIGger:BUS:B1:SPI:DATA:MISO:VALue.198
 :SEARCH:TRIGger:BUS:B1:SPI:DATA:MOSI:VALue.199
 :SEARCH:TRIGger:BUS:B1:CAN:CONDition199
 :SEARCH:TRIGger:BUS:B1:CAN:FRAMeType200
 SEARCH:TRIGger:BUS:B1:CAN:IDentifier:MODE ...201
 :SEARCH:TRIGger:BUS:B1:CAN:IDentifier:VALue...201
 :SEARCH:TRIGger:BUS:B1:CAN:IDentifier
 :DIRection202
 :SEARCH:TRIGger:BUS:B1:CAN:DATA:QUALifier ...203
 :SEARCH:TRIGger:BUS:B1:CAN:DATA:SIZE204
 :SEARCH:TRIGger:BUS:B1:CAN:DATA:VALue204
 :SEARCH:TRIGger:BUS:B1:LIN:CONDition205
 :SEARCH:TRIGger:BUS:B1:LIN:DATA:QUALifier206
 :SEARCH:TRIGger:BUS:B1:LIN:DATA:SIZE207
 :SEARCH:TRIGger:BUS:B1:LIN:DATA:VALue.....207
 :SEARCH:TRIGger:BUS:B1:LIN:ERRTYPE208
 :SEARCH:TRIGger:BUS:B1:LIN:IDentifier:VALue.....208
 :SEARCH:FFTPeak:METhod209
 :SEARCH:FFTPeak:METhod:MPeak209
 :SEARCH:FFTPeak:SINFo.....210

: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 →
		→ Query
:SEARCH:STATE		
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.	

		→ Query
:SEARCH:TOTAL		
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 →
		→ Query
:SEARCH:TRIGger:TYPE		
Description	Sets or queries the search trigger type.	
Syntax	:SEARCH:TRIGger:TYPE { EDGE PULSEWidth RUNT RISEFall FFTPeak BUS ? }	
Parameter/Return parameter	EDGE	Edge trigger
	PULSEWidth	Pulse width trigger
	RUNT	Runt trigger
	RISEFall	Rise and Fall trigger
	FFTPeak	FFT Peak trigger
	BUS	Bus trigger

Example :SEARCH:TRIGger:TYPe EDGE
Sets the search trigger to the edge type.

Set →

:SEARCH:TRIGger:SOURce → Query

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.

Set →

:SEARCH:TRIGger:EDGE:SLOP → Query

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 →

:SEARCH:TRIGger:LEVel → Query

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.

Set →



→ Query


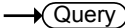
:SEARCH:TRIGger:LLEVel

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	
		

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  

Description Sets or queries the slope of the Rise and Fall search trigger.


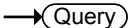
Syntax :SEARCH:TRIGger:RISEFall:SLOP { RISE | FALL | EITHER | ? }

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.

: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	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 | ?}

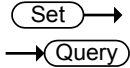
Related commands :SEARCH:TRIGger:TYPE
 :SEARCH:TRIGger:RUNT:TIME

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 (≠).



:SEARCH:TRIGger:RUNT:TIME

Description Sets or queries the pulse runt search trigger time.

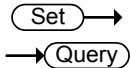
Syntax :SEARCH:TRIGger:RUNT:TIME {<NRf> | ? }

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.



: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

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 (≠).

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


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.

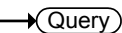
Example :SEARCH:TRIGger:RISEFall:TIME 4.00E-5
 Sets the trigger rise and fall time to 40.0us.

:SEARCH:TRIGger:BUS:TYPE → (Query)

Description	Returns the current bus type.	
Syntax	:SEARCH:TRIGger:BUS:TYPE?	
Return parameter	12C	I2C mode
	SPI	SPI mode
	UART	UART mode
	CAN	CAN mode
	LIN	LIN mode

Example :SEARCH:TRIGger:BUS:TYPE?
 UART


:SEARCH:TRIGger:BUS:B1:I2C:CONDition


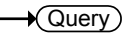


Description	Sets or queries the I ² C search trigger conditions.	
Syntax	:SEARCH:TRIGger:BUS:B1:I2C:CONDition {START STOP REPEATstart ACKMISS ADDRess DATA ADDRANDDATA ? }	
Parameter	START	Set Start as the I ² C search trigger condition.
	STOP	Set Stop as the I ² C search trigger condition.
	REPEATstart	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


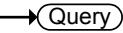
Description	Sets or queries the I ² C addressing mode (7 or 10 bits) for the search trigger.	
Syntax	:SEARCH:TRIGger:BUS:B1:I2C:ADDRess:MODE {ADDR7 ADDR10 ? }	

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:ADDRess: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 `<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 in binary.

Example 1 `:SEARCH:TRIGger:BUS:B1:I2C:ADDRes:MODE ADDR7`
`:SEARCH:TRIGger:BUS:B1:I2C:ADDRes:VALue "xxx0101"`
 Sets the address to XXX0101

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

`:SEARCH:TRIGger:BUS:B1:I2C:ADDRes`  →
`:DIRection` → 

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:ADDRes: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:ADDRes:DIRection READ Sets the direction to READ.	

Set →
 → Query

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 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 →
 → 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

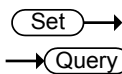
Return Parameter Returns the data value.

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


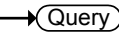


Description	Sets or queries the UART search triggering condition.	
Syntax	:SEARCH:TRIGger:BUS:B1:UART:CONDition { RXSTArt RXDATA RXENDPacket TXSTArt TXDATA TXENDPacket TXPARItYerr RXPARItYerr ? }	
Parameter	RXSTArt	Set search trigger on the RX Start Bit.
	RXDATA	Set search trigger on RX Data.
	RXENDPacket	Set search trigger on the RX End of Packet condition.

RXPARIttyerr	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.
TXPARIttyerr	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 (Set) →
→ (Query)

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.

Example 1 :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 (Set) →
→ (Query)

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: Set →
 VALue Query

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	<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: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) →

:SEARCH:TRIGger:BUS:B1:SPI:CONDition → (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.

(Set) →

:SEARCH:TRIGger:BUS:B1:SPI:DATA:SIZE → (Query)

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: Set →
VALue → Query

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> | ? }


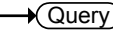
Related commands :SEARCH:TRIGger:BUS:B1:SPI:DATA:SIZE

Parameter	<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: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	<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 :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

 →

:SEARCH:TRIGger:BUS:B1:CAN:CONDition 

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.


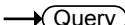
SEARCH:TRIGger:BUS:B1:CAN:IDentifier: 
MODE 

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 

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.


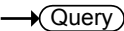
Syntax :SEARCH:TRIGger:BUS:B1:CAN:IDentifier:VALue {<string>|?}

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

Parameter/ Return parameter	<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: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 
```


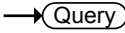
Description Sets or queries the address bit as read, write or don't care.

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
	NOCARE	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

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

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  

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
```

```
:SEARCH:TRIGger:BUS:B1:LIN:CONDition Set →
→Query
```

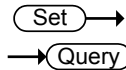
Description Sets or returns the LIN search trigger condition.

Syntax :SEARCH:TRIGger:BUS:B1:LIN:CONDition
 {SYNCFIELD|IDENTIFIER|DATA|IDANDDATA|WAKEUP|SLEEP|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.
	SLEEP	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:
QUALifier



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 (Set) →
→ (Query)

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 (Set) →
→ (Query)

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 →

:SEARCH:TRIGger:BUS:B1:LIN:ERRTYPE → 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

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

Set →

:SEARCH:FFTPeak:METHOD

→ 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

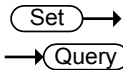
Set →

:SEARCH:FFTPeak:METHOD:MPEak

→ Query

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	<pre>: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</pre>	

: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>	


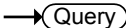
Label Commands

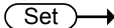
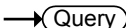
:CHANnel<X>:LABel.....	211
:CHANnel<X>:LABel:DISPlay.....	212
:REF<X>:LABel.....	212
:REF<X>:LABel:DISPlay.....	213
:BUS1:LABel.....	214
:BUS1:LABel:DISPlay.....	214
:SET<X>:LABel.....	215

: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	

		
		
: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	<pre>:CHANnel1:LABel "CH1" :CHANnel1:LABel:DISPlay ON :CHANnel1:LABel:DISPlay? ON</pre> <p>Sets the channel 1 label to "CH1" and then turns the label display on. The query return shows that the label is on.</p>	

		
		
: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.

Set →

→ Query

: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

Set →

→ Query

: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	<pre>:BUS1:LABel "Bus" :BUS1:LABel:DISPlay ON :BUS1:LABel:DISPlay? ON</pre> <p>Sets the label for the bus to "Bus" and then turns the label display on. The query return shows that the label is on.</p>	

:SET<X>:LABel

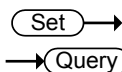
Description	Sets or returns the file label for the selected setup.	
Syntax	:SET<X>:LABel {<string> ?}	
Related commands	:SET<X>:LABel:DISPlay	
Parameter	<X>	Setup number 1 to 20
	<string>	The string must be no more than 8 characters and only contain alphanumeric characters in addition to period, dash 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	<pre>:SET1:LABel "SET1_lab"</pre> <p>Sets the label for setup 1 as "SET1_lab".</p>	

Example2 :SET1:LABel?
 SET1_lab

Segment Commands

:SEGMENTS:STATE	217
:SEGMENTS:CURRENT	218
:SEGMENTS:TOTALNUM	218
:SEGMENTS:TIME	219
:SEGMENTS:DISPALL	219
:SEGMENTS:MEASURE:MODE	219
:SEGMENTS:MEASURE:PLOT:SOURCE	220
:SEGMENTS:MEASURE:PLOT:DIVIDE	220
:SEGMENTS:MEASURE:PLOT:SELECT	221
:SEGMENTS:MEASURE:PLOT:RESULTS	221
:SEGMENTS:MEASURE:TABLE:SOURCE	222
:SEGMENTS:MEASURE:TABLE:SELECT	222
:SEGMENTS:MEASURE:TABLE:LIST	223
:SEGMENTS:MEASURE:TABLE:SAVE	223
:SEGMENTS:SAVE	223
:SEGMENTS:SAVE:SOURCE	224
:SEGMENTS:SAVE:SELECT:START	224
:SEGMENTS:SAVE:SELECT:END	225

: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:DISPALLSet →
→ 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:MODESet →
→ 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.

Set →
 → Query

:SEGMents:MEASure:PLOT:SOURce

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.

Set →
 → Query

:SEGMents:MEASure:PLOT:DIVide

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 GDS-2074E, serial number PXXXXXX, version V1.11",Segment Summary : CH1, Seg.,Pk- Pk (V),Pk-Pk (V),1,8.00m,8.00m.....etc

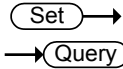
:SEGMents:MEASure:TABLE:SAVE Set →

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



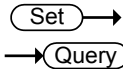
:SEGMents:SAVe:SOURce

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


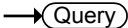
Description Sets or queries the starting segment to save from. The number of possible segments depends on the record length.

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	226
:DVM:SOURce	226
:DVM:MODE	227
:DVM:VALue	227

:DVM:STATE


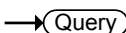




Description	Sets or queries the DVM state to on or off.	
Syntax	:DVM:STATE {OFF ON ? }	
Related commands	:DVM:SOURce :DVM:MODE	
Parameter/ Return parameter	OFF	Turns the DVM off.
	ON	Turns the DVM on.
Example	:DVM:STATE ON Turns the DVM state on.	

: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	CH1~CH4	Channel 1 to 4.
Example	:DVM:SOURce CH1 Sets the DVM source to channel 1.	

 											
: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	<table border="0"> <tr> <td>ACRMS</td> <td>Sets the mode to AC RMS</td> </tr> <tr> <td>DC</td> <td>Sets the mode to DC</td> </tr> <tr> <td>DCRMS</td> <td>Sets the mode to DC RMS</td> </tr> <tr> <td>DUTY</td> <td>Sets the mode to AC Duty</td> </tr> <tr> <td>FREQUENCY</td> <td>Sets the mode to AC frequency</td> </tr> </table>	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
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.										

	
: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.....	228
:GONogo:EXECute	228
:GONogo:FUNCTion.....	229
:GONogo:NGCount	229
:GONogo:NGDefine.....	229
:GONogo:SOURce	230
:GONogo:VIOLation	230
:GONogo:SCRipt.....	230
:TEMPlate:MODE	231
:TEMPlate:MAXimum.....	231
:TEMPlate:MINimum.....	231
:TEMPlate:POSition:MAXimum.....	232
:TEMPlate:POSition:MINimum	232
:TEMPlate:SAVe:MAXimum	232
:TEMPlate:SAVe:MINimum	233
:TEMPlate:TOLerance.....	233
:TEMPlate:SAVe:AUTo.....	233

:GONogo:CLear

Set →

Description Clears the Go/NoGo counter.

Syntax :GONogo:CLear

:GONogo:EXECute

Set →
→ Query

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:FUNctio (Set) →

Description Initializes the Go/NoGo APP. This must be run after the Go/NoGo APP has been started.

Syntax :GONogo:FUNctio

: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.

Set →
 → Query

: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.

Set →
 → Query

:GONogo:VIOLation	
Description	Sets or returns actions for the Go/NoGo violations.
Syntax	:GONogo:VIOLation {STOP CONTInue ?}
Parameter/ Return Parameter	STOP The waveform will be frozen. CONTInue Ignore the violation.
Example	:GONogo:VIOLation STOP Sets violation action to STOP.

Set →

:GONogo:SCRipt	
Description	Activates/Deactivates the Go/NoGo APP or queries its state.
Syntax	:GONogo:SCRipt {OFF ON ?}
Parameter/ Return Parameter	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	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	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	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%.

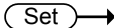
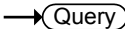
:TEMPlate:SAVe:AUTO (Set) →



Description Saves the AUTO template (maximum and minimum templates).

Syntax :TEMPlate:SAVe:AUTO

Data Logging Commands

:DATALOG:STATE	234
:DATALOG:SOURce	234
:DATALOG:SAVe	235
:DATALOG:INTerval	235
:DATALOG:DURation	236

:DATALOG:STATE		 
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.	

:DATALOG:SOURce		 
Description	Sets or queries the data logging source channel.	
Syntax	:DATALOG:SOURce {CH1~CH4 all ?}	
Related commands	:DATALOG:STATE :DATALOG:SAVe :DATALOG:INTerval :DATALOG:DURation	
Parameter/Return	CH1 ~CH4	Channel 1, 2, 3 or 4

parameter 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

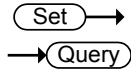
Description Sets or queries the recording interval time in seconds. The interval times that can be set are dependent on the settings of the DATALOG:SOURCE and DATALOG:SAVE commands.

Syntax :DATALOG:INTerval {<NR1>|?}

Related commands :DATALOG:STATE
 :DATALOG:SOURce
 :DATALOG:SAVe
 :DATALOG:DURation

Parameter/Return parameter	<NR1>	<p>Sets returns the interval time in discrete seconds:</p> <p>Interval time for DATALOG:SOURCE = All or DATALOG:SAVE=IMAGE: 5, 10, 15, 20, 25, 30, 35, 60, 120</p> <p>Interval time for DATALOG:SOURCE = CH1~CH4: 2, 3, 4, 5, 10, 20, 30, 60, 120</p>
----------------------------	-------	---

Example : DATALOG:INTerval 5
 Sets the recording interval to 5 seconds.



:DATALOG:DURation

Description	Sets or queries the recording duration time in minutes.
Syntax	:DATALOG:DURation {<NR1> ?}
Related commands	:DATALOG:STATE :DATALOG:SOURce :DATALOG:SAVe :DATALOG:INTerval

Parameter/Return parameter	<NR1>	<p>Sets returns the duration time in discrete minutes:</p> <p>5, 10, 15, 20, 25, 30, 60, 90, 120, 150, 180, 210, 240, 270, 300, 330, 360, 390, 420, 450, 480, 510, 540, 570, 600, 1200, 1800, 2400, 3000, 3600, 4200, 4800, 5400, 6000</p>
----------------------------	-------	--

Example : DATALOG:DURation 10
 Sets the recording duration to 10 minutes.

Remote Disk Commands

:REMOTEDisk:IPADdress	237
:REMOTEDisk:PATHName	237
:REMOTEDisk:USERName	237
:REMOTEDisk:PASSWord	238
:REMOTEDisk:MOUNT.....	238
:REMOTEDisk:AUTOMount.....	239

: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.	

:REMOTEDisk:USERName
 Set →
 Query

Description	Sets or queries the account username for the remote disk.	
Syntax	:REMOTEDisk:USERName {<string> ? }	

Parameter/Return parameter	<code><string></code>	User name enclosed in double quotes eg., "User_Name".
----------------------------	-----------------------------	---

Example :REMOTEDisk:USERName "User_Name"
Sets the account name as User_Name.

(Set) →

:REMOTEDisk:PASSWord

→ (Query)

Description	Sets or queries the account password for the remote disk.
-------------	---

Syntax	:REMOTEDisk:PASSWord {<string> ? }
--------	--------------------------------------

Parameter/Return parameter	<code><string></code>	Username password enclosed in double quotes eg., "Password".
----------------------------	-----------------------------	--

Example :REMOTEDisk:PASSWord "Password"
Sets the account password as Password.

(Set) →

:REMOTEDisk:MOUNT

→ (Query)

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.	

APPENDIX

Error messages

Description The following error messages may be returned from the :SYSTem:ERRor? query. For details see page 155.

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
AUTORSET	
MODE.....	45
AUTOSet	45
BUS1	164
CAN	
BITRate	173
PROBe.....	173
SAMPLepoint.....	173
SOURce.....	172
DISplay	
FORMAt.....	172
I2C	
ADDRess	
RWINclude	165
SCLK	
SOURce.....	166
SDA	
SOURce.....	166
INPut	165
LABel.....	214
DISPlay.....	214
LIN	
BITRate	174
IDFORmat	174
POLARity.....	175
SAMPLepoint.....	175
SOURce.....	175
STANDard	176
SPI	
BITORder	170
MISO	
SOURce.....	171
MOSI	
SOURce.....	171
SCLK	
POLARity	169
SOURce.....	170
SS	
POLARity	169
SOURce.....	171
WORDSize	170
STATE	164
TYPE.....	164
UART	
BITRate	166
DATABits.....	167
EOFPACKET.....	168
PACKEt	168
PARity	167
RX	
SOURce.....	169
TX	
SOURce.....	168
CHANnel	
BWLlimit	46
COUPLing	47
DESKew	47
DISPlay.....	47
EXPand.....	48
IMPedance.....	48
INVert.....	49

LABel.....	211	DATE.....	162
DISPlay.....	212	DISPlay	
POStion.....	49	INTENSity	
PROBE		BACKLight	
TYPE.....	50	AUTODim	
PROBE		TIME.....	73
RATio.....	50	DISPlay	
SCALe.....	50	GRATicule.....	74
CLS.....	34	INTensity	
CURSor		BACKLight.....	72
DDT.....	64	BACKLight	
H1Position.....	65	AUTODim	
H2Position.....	65	ENABLE.....	72
HDELta.....	66	GRATicule.....	72
HUNI.....	63	WAVEform.....	71
HUSE.....	63	OUTPut.....	74
MODE.....	62	PERSistence.....	73
SOURce.....	62	WAVEform.....	74
V1Position.....	66	DVM	
VDELta.....	67	MODE.....	227
VUNI.....	64	SOURce.....	226
VUSE.....	64	STATE.....	226
XY		VALue.....	227
POLar		Error list.....	240
RADIUS		Ethernet	
DELta.....	69	interface.....	7
POStion.....	68	ETHERnet	
THETA		DHCP.....	161
DELta.....	69	FORCe.....	114
POStion.....	69	GONogo	
PRoDuct		CLEar.....	228
DELta.....	70	EXECute.....	228
POStion.....	70	FUNctioN.....	229
RATio		NGCount.....	229
DELta.....	71	NGDefine.....	229
RATio		SCRipt.....	230
POStion.....	70	SOURce.....	230
RECTangular		VIOLation.....	230
X		HARDcopy	
DELta.....	67	ASSIGN.....	77
POStion.....	67	MODE.....	75
Y		PRINtINKSaver.....	76
DELta.....	68	SAVEFORMat.....	77
POStion.....	68	SAVEINKSaver.....	76
DATALOG			
DURation.....	236		
INTerval.....	235		
SAVE.....	235		
SOURce.....	234		
STATE.....	234		

START	75
HEADer	44
IDN?	32
Interface	5
LISTer	
DATA	172
LRN?	32
MARK	177
CREATE	177
DELEte	178
MATH	
ADVanced	
OPERator	57
POSition	59
SCALE	59
DISP	52
DUAL	
OPERator	53
POSition	54
SCALE	54
SOURce	53
FFT	
HORizontal	
POSition	57
SCALE	57
MAG	55
POSition	56
SCALE	56
SOURce	55
WINDow	55
TYPE	53
MATHVAR	58
MEASure	
AMPlitude	90
AREa	96
CARea	96
CMEan	91
CRMS	95
FALL	82
FFFDelay	99
FFRDelay	98
FOVShoot	82
FPReshoot	83
FREQuency	83
FRFDelay	97
FRRDelay	97

GATing	79
HIGH	92
LFFDelay	101
LRFDelay	100
LOW	92
LRFDelay	100
LRRDelay	99
MAX	93
MEAN	90
METHod	80
MIN	93
NEDGE	89
NPULSE	88
NWIDth	84
PDUTy	84
PEDGE	89
PERiod	85
PHase	102
PK2PK	94
PPULSE	88
PWIDth	85
RISe	86
RMS	94
ROVShoot	86
RPReshoot	87
SOURce	79

MEASUREMENT

MEAS	
MAXimum	106
MEAN	107
MINimum	107
SOURCE	103
STATE	104
STDdev	108
TYPe	104
VALue	105
REFLevel	
PERCent	
HIGH	80
LOW	81
MID	81
MID2	81
STATIstics	109
MODE	109
WEIghting	109

RCL	33
-----------	----

RECALL

SETUp	156
WAVEform	156

REF	DIRection.....	202
DISPlay.....	MODE.....	201
LABEl.....	VALue.....	201
DISPlay.....	I2C	
OFFSet.....	ADDRESS	
SCALE.....	DIRection.....	191
TIMEbase	MODE.....	189
POSition.....	TYPE.....	190
SCALE.....	VALue.....	191
Remote control	CONDition.....	189
interface configuration.....	DATA	
.....	SIZE.....	192
.....	VALue.....	192
REMOVEDisk	LIN	
AUTOMount.....	CONDition.....	205
IPADDRESS.....	DATA	
MOUNT.....	QUALifier.....	206
PASSWORD.....	SIZE.....	207
PATHName.....	VALue.....	207
USERName.....	ERRTYPE.....	208
RST.....	IDentifier	
RUN.....	VALue.....	208
SAV.....	SPI	
SAVE	CONDition.....	197
IMAGE.....	DATA	
FILEFormat.....	MISO.....	198
INKSaver.....	MOSI.....	199
SETUp.....	SIZE.....	197
WAVEform.....	UART	
FILEFormat.....	CONDition.....	193
SEARCH	RX	
COPY.....	DATA.....	194, 195
FFTPeak	TX	
METHod.....	DATA.....	195, 196
MPEak.....	TYPE.....	188
SINFo.....	EDGE	
STATE.....	SLOP.....	182
TOTAL.....	HLEVel.....	183
TRIGger	LEVel.....	182
BUS	LLEVel.....	183
B1	PULSE	
CAN	TIME.....	186
CONDition.....	WHEN.....	185
DATA	PULSEWidth	
QUALifier.....	POLarity.....	184
SIZE.....	RISEFall	
VALue.....	SLOP.....	185
FRAMetype.....	TIME.....	188
IDentifier	WHEN.....	187
	RUNT	
	POLarity.....	184
	TIME.....	187
	WHEN.....	186
	SOURCE.....	182
	TYPE.....	181
	SEGMENTS	

CURRent	218	MODe	116
DISPALL	219	POSition	115
MEASure		SCALe.....	115
MODe	219	WINDow	
PLOT		POSition.....	116
DIVide.....	220	SCALe.....	117
RESults.....	221		
SElect.....	221	TRIGger	
SOURce.....	220	ALTernate.....	133
TABLE		BUS	
LIST	223	B1	
SAVe.....	223	CAN	
SElect.....	222	CONDition	145
SOURce.....	222	DATA	
SAVE	223	QUALifier	148
SElect		SIZE	149
END.....	225	VALue	149
START.....	224	FRAMetype.....	146
SOURce.....	224	IDentifier	
STATE	217	DIRection	148
TIME.....	219	MODE	146
TOTALnum	218	VALue	147
SET		I2C	
LABel	215	ADDRESS	
SINGLE.....	113	DIRection	138
Socket server		MODE	136
function check	10	TYPE.....	137
Socket server		VALue	137
interface.....	9	CONDition	136
STOP	113	DATA	
SYSTEM		SIZE	139
ERRor	155	VALue	139
LOCK.....	155	LIN	
TEMPLATE		CONDition	150
MAXimum.....	231	DATA	
MINimum.....	231	QUALifier	151
MODE.....	231	SIZE	152
POSition		VALue	152
MAXimum	232	ERRTYPE	153
MINimum	232	IDentifier	
SAVE		VALue	153
AUTO	233	SPI	
MAXimum	232	CONDition	143
MINimum	233	DATA	
TOLerance	233	MISO	
TIMEbase		VALue.....	144
EXPand.....	115	MOSI	
		VALue.....	145
		SIZE	143
		UART	
		CONDition	140
		RX	
		DATA	
		SIZE	140
		VALue.....	141
		TX	

DATA	132
SIZE.....	142
VALue	142
THReshold	
CH	135
TYPe	135
COUPle	121
DELay	
EVENt	125
LEVel	126
SLOP.....	124
TIME.....	125
TYPe.....	125
EDGE	
SLOP.....	124
EXTERnal	
PRObe	
RATio.....	134
TYPe.....	134
FREQuency.....	120
HLEVel.....	123
HOLDoff.....	122
LEVel.....	122
LLEVel.....	123
MODE.....	122
NREJ	121
PULSe	
TIME.....	132
WHEN.....	131
PULSEWidth	
POLarity.....	126
RISEFall	
SLOP.....	128
TIME.....	129
WHEN.....	128
RUNT	
POLarity.....	127
TIME.....	128
WHEN.....	127
SOURce	121
STATe.....	133
TIMEOut	
TIMER.....	133
WHEN.....	132
TYPe	120
VIDeo	
FIELD.....	130
LINE.....	130
POLarity.....	131
TYPe.....	129
USB	
function check.....	6
remote control interface	5