

# Digital Oscilloscope

GDS-3000A series

---

## PROGRAMMING MANUAL



ISO-9001 CERTIFIED MANUFACTURER

**GW INSTEK**

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.*

*NIFM, NI-VIS/ATM are trademarks of National Instruments Corporation*

*Product and company names mentioned herein are trademarks or trade names of their respective companies.*

**Good Will Instrument Co., Ltd.**

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

---

---

# Table of Contents

<b>INTERFACE OVERVIEW .....</b>	<b>5</b>
Interface Configuration .....	5
<b>COMMAND OVERVIEW .....</b>	<b>15</b>
Command Syntax .....	15
List of Commands in Functional Order .....	17
<b>COMMAND DETAILS .....</b>	<b>42</b>
Common Commands .....	44
Acquisition Commands .....	50
Autoscale Commands .....	59
Vertical Commands .....	60
Math Commands .....	66
Cursor Commands .....	77
Display Commands .....	90
Hardcopy Commands .....	96
Measure Commands .....	99
Measurement Commands .....	127
Reference Commands .....	136
Run Command .....	139
Stop Command .....	139
Single Command .....	139
Force Command .....	140
Timebase Commands .....	141
Trigger Commands .....	144
System Commands .....	186
Save/Recall Commands .....	187
Ethernet Commands .....	192
Time Command .....	196
Bus Decode Commands .....	197
Mark Commands .....	213

Mask Commands .....	215
FRA Commands .....	230
Search Commands .....	238
Label Commands .....	277
Segment Commands .....	285
DVM Commands.....	296
Go_NoGo Commands .....	298
AWG Commands.....	304
Data Logging Commands.....	326
Remote Disk Commands.....	329
Spectrum Analyzer Commands .....	332
Power Analysis Commands .....	349
USB Delay Command .....	414
Digital Commands .....	415
<b>APPENDIX.....</b>	<b>423</b>
Error messages .....	423
<b>INDEX.....</b>	<b>427</b>

# INTERFACE OVERVIEW

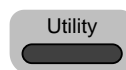
This manual describes how to use the GDS-3000A's remote command functionality and lists the command details. The Overview chapter describes how to configure the USB and Ethernet remote control interfaces.

## Interface Configuration

### Configure USB Interface

USB Configuration	PC side connector	Type A, host
	GDS-3000A side connector	Type B, device
	Speed	1.1/2.0
	USB Class	USBTMC 488.2 class device for remote connectivity

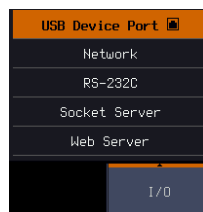
Panel Operation 1. Press the *Utility* key.



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



3. Rotate the "VARIABLE" knob to select the *USB Device Port* function.



4. Select *Computer* from the side menu.



5. This oscilloscope is a USB-TMC device. Please install the National Instruments NI-VISA library which can download from the National Instruments web site. Newer versions are likely, and should be compatible with this instrumentation. Download the latest version available for the operating system being used by the controlling computer.



## Configure the Ethernet Interface

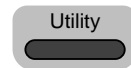
Ethernet Configuration	MAC Address	Domain Name
	Instrument Name	DNS IP Address
	User Password	Gateway IP Address
	Instrument IP Address	Subnet Mask

Background	The Ethernet interface is used for remote control using a socket server connection.
------------	---

1. Connect the Ethernet cable to the LAN port on the rear panel.



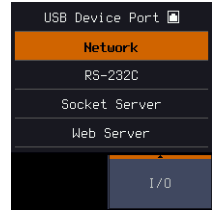
2. Press the *Utility* key.



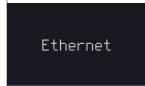
3. Press *I/O* from the bottom menu.



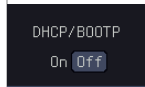
4. Rotate the “VARIABLE” knob to select the *Network* function.



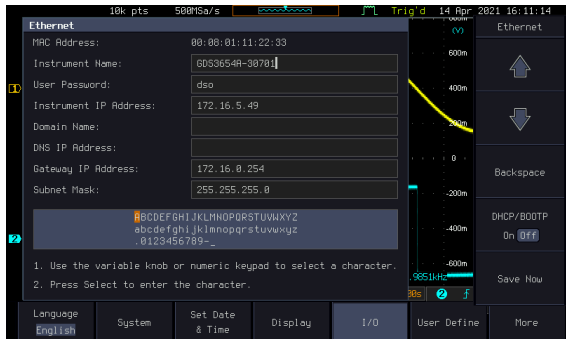
5. Press Ethernet from the side menu



6. Set DHCP/BOOTP to On or Off from the side menu.



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

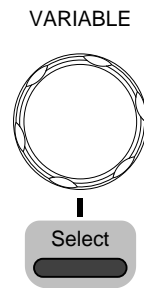


- Use the Up and Down arrows on the side menu or use the numerical keypad on front panel to navigate to each Ethernet configuration item.



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

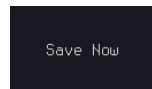
- Use the *Variable* knob to highlight a character and use the *Select* key to choose a character.



- Press *Backspace* to delete a character.



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





## Configure Socket Server

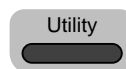
The GDS-3000A supports socket server functionality for direct two-way communication with a client PC or device over LAN. By default, the Socket Server is off.

Configure Socket Server

1. Configure the IP address for the GDS-3000A.

Page 6

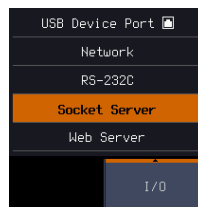
2. Press the *Utility* key.



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



4. Rotate the "VARIABLE" knob to select the *Socket Server* function.

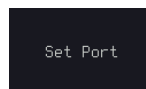


5. Press *Select Port* and choose the port number with the Variable knob.

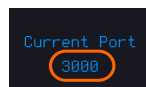


Range 1024~65535

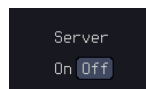
6. Press *Set Port* to confirm the port number.



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



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



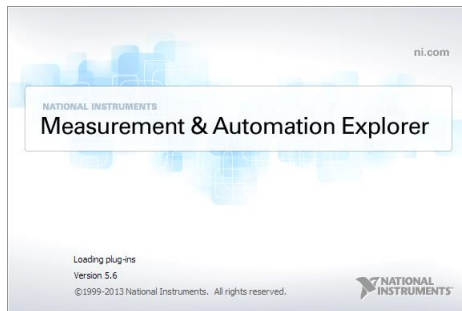
## Socket Server Functionality Check

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](http://www.ni.com).

- Operation
1. Configure the IP address for the GDS-3000A. Page 6
  2. Configure the socket port.
  3. Start the NI Measurement and Automation Explorer (MAX) program. Using Windows, press:

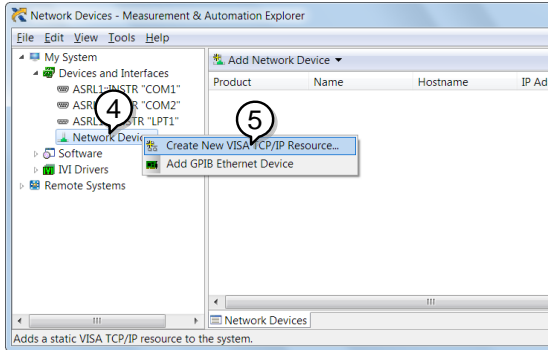


Start>All Programs>National Instruments>Measurement & Automation

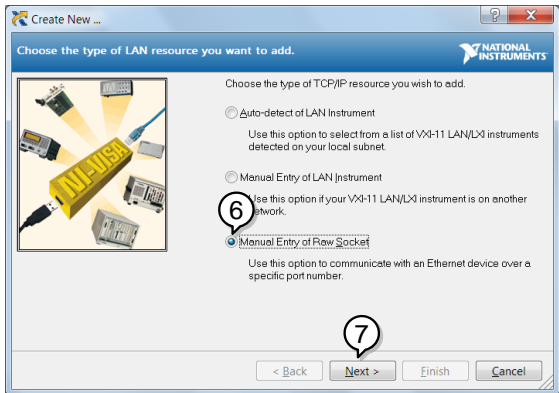


4. From the Configuration panel access;
 

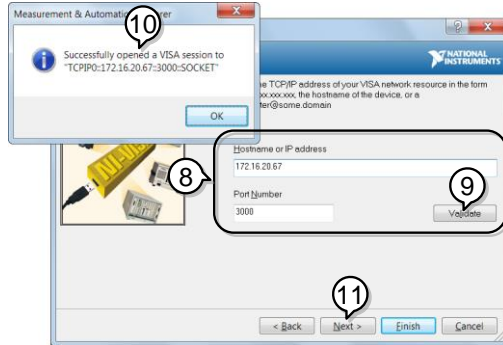
My System>Devices and Interfaces>Network Devices
5. Right click Network Devices and select Create New Visa TCP/IP Resource...



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

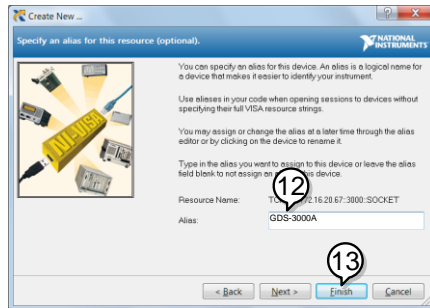


8. Enter the GDS-3000A's IP address and socket port number.
9. Click Validate.
10. A popup will appear to tell you if a VISA socket session was successfully created.
11. Click *Next*.



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

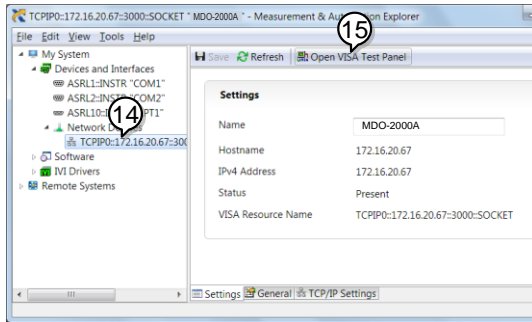
13. Click *Finish* to finish the configuration.



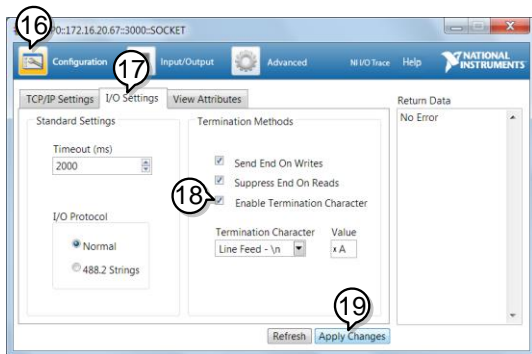
14. The GDS-3000A will now appear under Network Devices in the Configuration Panel.

Functionality  
Check

- Click the *Open Visa Test Panel* to send a remote command to the GDS-3000A.

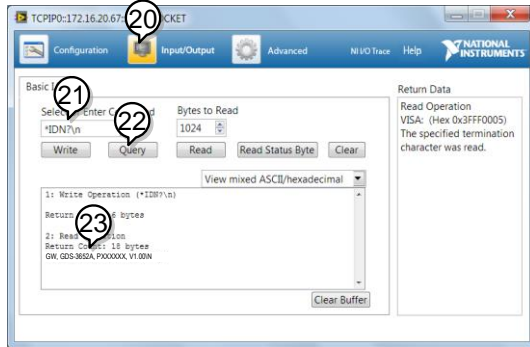


- Click on the *Configuration* icon.
- Select the *I/O Settings* tab.
- Mark the *Enable Termination Character* checkbox. Make sure the termination character is a line feed (/n, value: xA).
- Click *Apply Changes*.



- Click the *Input/Output* icon.
- Make sure the *\*IDN?* query is selected in the *Select or Enter Command* drop box.
- Click on *Query*.

23. The manufacturer, model number, serial number and firmware version will be displayed in the buffer. For example: GW-INSTEK, GDS-3652A,PXXXXXX,V1.00



# COMMAND OVERVIEW

The Command overview chapter lists all GDS-3000A 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

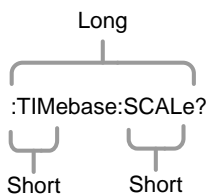
---

Compatible standard

- USBTMC 488.2 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 :TIMbase: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 commands	*IDN? .....	44
	*LRN? .....	44
	*SAV .....	45
	*RCL .....	45
	*RST .....	46
	*CLS .....	46
	*ESE .....	46
	*ESR .....	47
	*OPC .....	47
	*SRE .....	48
	*STB .....	49
Acquisition commands	:ACQuire:AVERage .....	50
	:ACQuire:MODE .....	51
	:ACQuire<X>:MEMory? .....	51
	:ACQuire:FILTer:SOURce .....	53
	:ACQuire:FILTer .....	54
	:ACQuire:FILTer:FREQuency .....	54
	:ACQuire:FILTer:FREQuency:UPPER .....	55
	:ACQuire:FILTer:FREQuency:LOWER .....	55
	:ACQuire:FILTer:TYPe .....	55
	:ACQuire:FILTer:TRACking .....	56
	:ACQuire<X>:STATE? .....	56
	:ACQuire:INTERpolation .....	57
	:ACQuire:RECOrdlength .....	57
	:HEADer .....	57
:ACQuire:SAMPlerate? .....	58	
Autoscale commands	:AUTOSet .....	59
	:AUTORSET:MODE .....	59
Vertical Scale commands	:CHANnel<X>:BWLimit .....	60
	:CHANnel<X>:COUPling .....	61

	:CHANnel<X>:DESKew .....	61
	:CHANnel<X>:DISPlay .....	61
	:CHANnel<X>:EXPand .....	62
	:CHANnel<X>:IMPedance? .....	62
	:CHANnel<X>:INVert .....	63
	:CHANnel<X>:POSition .....	63
	:CHANnel<X>:PROBe:RATio .....	64
	:CHANnel<X>:PROBe:TYPe .....	64
	:CHANnel<X>:SCALe .....	65
<hr/>		
Math commands	:MATH:DISP .....	66
	:MATH:TYPe .....	67
	:MATH:DUAL:SOURce<X> .....	69
	:MATH:DUAL:OPERator .....	70
	:MATH:DUAL:POSition .....	70
	:MATH:DUAL:SCALe .....	70
	:MATH:FFT:SOURce .....	71
	:MATH:FFT:MAG .....	71
	:MATH:FFT:WINDow .....	72
	:MATH:FFT:POSition .....	72
	:MATH:FFT:SCALe .....	72
	:MATH:FFT:HORizontal:SCALe .....	73
	:MATH:FFT:HORizontal:POSition .....	73
	:MATH:DEFine .....	73
	:MATHVAR? .....	75
	:MATHVAR:VAR<X> .....	75
	:MATH:ADVanced:POSition .....	75
	:MATH:ADVanced:SCALe .....	76
<hr/>		
Cursor commands	:CURSor:MODE .....	78
	:CURSor:SOURce .....	78
	:CURSor:HUNI .....	79
	:CURSor:HUSE .....	79
	:CURSor:VUNI .....	80
	:CURSor:VUSE .....	80
	:CURSor:DDT .....	80

	:CURSor:H1Position .....	81
	:CURSor:H2Position .....	81
	:CURSor:HDELta .....	82
	:CURSor:V1Position.....	82
	:CURSor:V2Position.....	82
	:CURSor:VDELta .....	83
	:CURSor:XY:RECTangular:X:POSition<X> .....	83
	:CURSor:XY:RECTangular:X:DELta .....	83
	:CURSor:XY:RECTangular:Y:POSition<X> .....	84
	:CURSor:XY:RECTangular:Y:DELta .....	84
	:CURSor:XY:POLar:RADIUS:POSition<X> .....	84
	:CURSor:XY:POLar:RADIUS:DELta .....	85
	:CURSor:XY:POLar:THETA:POSition<X> .....	85
	:CURSor:XY:POLar:THETA:DELta .....	85
	:CURSor:XY:PRODuct:POSition<X> .....	86
	:CURSor:XY:PRODuct:DELta.....	86
	:CURSor:XY:RATio:POSition<X> .....	86
	:CURSor:XY:RATio:DELta .....	87
	:CURSor:HTRACKing .....	87
	:CURSor:VTRACKing .....	87
	:CURSor:MARK .....	88
	:CURSor:SA:SOURce .....	88
Display	:DISPlay:INTensity:WAVEform.....	90
commands	:DISPlay:INTensity:GRATICule.....	90
	:DISPlay:INTensity:BACKLight .....	91
	:DISPlay:INTensity:BACKLight:AUTODim:ENABLE ...	91
	:DISPlay:INTENSITY:BACKLight:AUTODim:TIME ...	91
	:DISPlay:PERSistence .....	92
	:DISPlay:GRATICule .....	92
	:DISPlay:WAVEform.....	93
	:DISPlay:OUTPut .....	93
	:DISPlay:PNGOutput? .....	93
	:DISPlay:WAVEform:COLor.....	94
	:DISPlay:RULER .....	94

---

	:DISPlay:TRANSREADouts .....	94
Hardcopy commands	:HARDcopy:START .....	96
	:HARDcopy:MODE .....	96
	:HARDcopy:PRINTINKSaver .....	97
	:HARDcopy:SAVEINKSaver .....	97
	:HARDcopy:SAVEFORMat .....	97
	:HARDcopy:ASSIGN.....	98
Measure commands	:MEASure:GATing.....	100
	:MEASure:SOURce<X> .....	101
	:MEASure:METHod .....	101
	:MEASurement:REFLevel:PERCent:HIGH .....	102
	:MEASurement:REFLevel:PERCent:LOW .....	102
	:MEASurement:REFLevel:PERCent:MID .....	102
	:MEASurement:REFLevel:PERCent:MID2 .....	103
	:MEASurement:REFLevel:ABSolute:HIGH.....	103
	:MEASurement:REFLevel:ABSolute:LOW .....	103
	:MEASurement:REFLevel:ABSolute:MID .....	104
	:MEASurement:REFLevel:ABSolute:MID2 .....	104
	:MEASurement:REFLevel:METHod .....	104
	:MEASure:FALL.....	105
	:MEASure:FOVShoot.....	105
	:MEASure:FPReshoot .....	106
	:MEASure:FREQuency .....	107
	:MEASure:NWIDth .....	107
	:MEASure:PDUTy .....	108
	:MEASure:PERiod .....	108
	:MEASure:PWIDth .....	109
	:MEASure:RISe .....	109
	:MEASure:ROVShoot.....	110
	:MEASure:RPReshoot.....	110
	:MEASure:PPULSE.....	111
	:MEASure:NPULSE.....	112
	:MEASure:PEDGE .....	112
	:MEASure:NEDGE .....	113

---

	:MEASure:AMPlitude.....	113
	:MEASure:MEAN.....	114
	:MEASure:CMEan.....	114
	:MEASure:HIGH.....	115
	:MEASure:LOW.....	115
	:MEASure:MAX.....	116
	:MEASure:MIN.....	117
	:MEASure:PK2PK.....	117
	:MEASure:RMS.....	118
	:MEASure:CRMS.....	118
	:MEASure:AREa.....	119
	:MEASure:CARea.....	119
	:MEASure:FRRDelay.....	120
	:MEASure:FRFDelay.....	121
	:MEASure:FFRDelay.....	121
	:MEASure:FFFDelay.....	122
	:MEASure:LRRDelay.....	122
	:MEASure:LRFDelay.....	123
	:MEASure:LFRDelay.....	124
	:MEASure:LFFDelay.....	124
	:MEASure:PHase.....	125
	:MEASure:PFLicker.....	125
	:MEASure:FLicker.....	126
Measurement	:MEASUrement:MEAS<X>:SOURCE<X>.....	127
commands	:MEASUrement:MEAS<X>:TYPE.....	128
	:MEASUrement:MEAS<X>:STATE.....	129
	:MEASUrement:MEAS<X>:VALue.....	129
	:MEASUrement:MEAS<X>:MAXimum.....	130
	:MEASUrement:MEAS<X>:MEAN.....	131
	:MEASUrement:MEAS<X>:MINImum.....	131
	:MEASUrement:MEAS<X>:STDdev.....	132
	:MEASUrement:STATIstics:MODE.....	133
	:MEASUrement:STATIstics:WEIghting.....	133
	:MEASUrement:STATIstics.....	133

	:MEASUrement:INDICators:STAT .....	134
	:MEASUrement:INDICators:HORIZ<x>? .....	134
	:MEASUrement:INDICators:VERT<x>? .....	135
	:MEASUrement:INDICators:NUMHORZ? .....	135
	:MEASUrement:INDICators:NUMVERT? .....	135
<hr/>		
Reference	:REF<X>:DISPlay .....	136
commands	:REF<X>:TIMebase:POSition .....	136
	:REF<X>:TIMebase:SCALe .....	137
	:REF<X>:OFFSet .....	137
	:REF<x>:SCALe .....	138
<hr/>		
Run command	:RUN .....	139
<hr/>		
Stop command	:STOP .....	139
<hr/>		
Single command	:SINGle .....	139
<hr/>		
Force command	:FORCe .....	140
<hr/>		
Timebase	:TIMebase:EXPand .....	141
commands	:TIMebase:POSition .....	141
	:TIMebase:SCALe .....	142
	:TIMebase:MODE .....	142
	:TIMebase:WINDow:POSition .....	142
	:TIMebase:WINDow:SCALe .....	143
	:PLAYStop .....	143
<hr/>		
Trigger	:TRIGger:FREQUency .....	146
commands	:TRIGger:TYPe .....	146
	:TRIGger:SOURce .....	147
	:TRIGger:COUPLe .....	147
	:TRIGger:NREJ .....	148
	:TRIGger:MODE .....	148
	:TRIGger:HOLDoff .....	148
	:TRIGger:LEVel .....	149

---

:TRIGger:HLEVel.....	149
:TRIGger:LLEVel.....	150
:TRIGger:EDGe:SLOP .....	150
:TRIGger:DELaY:SLOP .....	151
:TRIGger:DELaY:TYPe .....	151
:TRIGger:DELaY:TI梅.....	151
:TRIGger:DELaY:EVENt.....	152
:TRIGger:DELaY:LEVel .....	152
:TRIGger:PULSEWidth:POLarity.....	152
:TRIGger:RUNT:POLarity .....	153
:TRIGger:RUNT:WHEn .....	153
:TRIGger:RUNT:TI梅.....	154
:TRIGger:RISEFall:SLOP .....	154
:TRIGger:RISEFall:WHEn.....	155
:TRIGger:RISEFall:TI梅.....	155
:TRIGger:VIDeo:TYPe .....	156
:TRIGger:VIDeo:FIELD .....	156
:TRIGger:VIDeo:LINE.....	157
:TRIGger:VIDeo:POLarity .....	157
:TRIGger:PULSe:WHEn .....	157
:TRIGger:PULSe:TI梅.....	158
:TRIGger:TIMEOut:WHEn .....	158
:TRIGger:TIMEOut:TIMER .....	159
:TRIGger:ALTErnate .....	159
:TRIGger:STATE .....	160
:TRIGger:EXTErNal:PROBe:TYPe.....	160
:TRIGger:EXTErNal:PROBe:RATio .....	161
:TRIGger:BUS:TYPe .....	161
:TRIGger:BUS:THReshold:CH<x> .....	162
:TRIGger:BUS:B1:I2C:CONDition .....	162
:TRIGger:BUS:B1:I2C:ADDReSS:MODe.....	163
:TRIGger:BUS:B1:I2C:ADDReSS:TYPe.....	163
:TRIGger:BUS:B1:I2C:ADDReSS:VALue .....	164
:TRIGger:BUS:B1:I2C:ADDReSS:DIRectioN.....	165
:TRIGger:BUS:B1:I2C:DATA:SIZE .....	165

---

:TRIGger:BUS:B1:I2C:DATA:VALue .....	166
:TRIGger:BUS:B1:UART:CONDition .....	166
:TRIGger:BUS:B1:UART:RX:DATA:SIZE .....	167
:TRIGger:BUS:B1:UART:RX:DATA:VALue .....	168
:TRIGger:BUS:B1:UART:TX:DATA:SIZE .....	168
:TRIGger:BUS:B1:UART:TX:DATA:VALue .....	169
:TRIGger:BUS:B1:SPI:CONDition .....	170
:TRIGger:BUS:B1:SPI:DATA:SIZE .....	170
:TRIGger:BUS:B1:SPI:DATA:MISO:VALue .....	171
:TRIGger:BUS:B1:SPI:DATA:MOSI:VALue .....	171
:TRIGger:BUS:B1:CAN:CONDition.....	172
:TRIGger:BUS:B1:CAN:FRAMetype .....	173
:TRIGger:BUS:B1:CAN:IDentifier:MODE.....	173
:TRIGger:BUS:B1:CAN:IDentifier:VALue .....	174
:TRIGger:BUS:B1:CAN:IDentifier:DIRection .....	174
:TRIGger:BUS:B1:CAN:DATA:QUALifier .....	175
:TRIGger:BUS:B1:CAN:DATA:SIZE.....	176
:TRIGger:BUS:B1:CAN:DATA:VALue.....	176
:TRIGger:BUS:B1:LIN:CONDition .....	177
:TRIGger:BUS:B1:LIN:DATA:QUALifier .....	178
:TRIGger:BUS:B1:LIN:DATA:SIZE .....	178
:TRIGger:BUS:B1:LIN:DATA:VALue .....	179
:TRIGger:BUS:B1:LIN:ERRTYPE.....	180
:TRIGger:BUS:B1:LIN:IDentifier:VALue .....	180
:TRIGger:BUS:B1:PARallel:VALue .....	181
:TRIGger:LOGic:INPut:CLOCK:SOURce.....	181
:TRIGger:LOGic:INPut:CLOCK:EDGE.....	182
:TRIGger:LOGic:FUNcTION .....	182
:TRIGger:LOGic:PATtern .....	183
:TRIGger:LOGic:PATtern:INPut:D<x> .....	183
:TRIGger:LOGic:PATtern:DELtatime.....	184
:TRIGger:LOGic:PATtern:WHEN .....	184

System	:SYSTem:LOCK.....	186
commands	:SYSTem:ERRor .....	186



Save/Recall commands	:RECALL:SETUp .....	187
	:RECALL:WAVEform .....	187
	:SAVe:IMAGe .....	188
	:SAVe:IMAGe:FILEFormat .....	188
	:SAVe:IMAGe:INKSaver .....	189
	:SAVe:SETUp .....	189
	:SAVe:WAVEform .....	190
	:SAVe:WAVEform:FILEFormat .....	191
<hr/>		
Ethernet Commands	:ETHERnet:DHCP .....	192
	:ETHERnet:IPADdress .....	192
	:ETHERnet:DNS:IPADdress .....	193
	:ETHERnet:DOMAINname .....	193
	:ETHERnet:ENET:ADDress? .....	193
	:ETHERnet:GATEWay:IPADdress .....	193
	:ETHERnet:NAME .....	194
	:ETHERnet:PASSWord .....	194
	:ETHERnet:SUBNETMask .....	194
<hr/>		
Time Command	:DATE .....	196
<hr/>		
Bus Decode Commands	:BUS1 .....	198
	:BUS1:STATE .....	198
	:BUS1:TYPE .....	198
	:BUS1:INPut .....	199
	:BUS1:I2C:ADDress:RWINClude .....	199
	:BUS1:I2C:SCLK:SOURce .....	200
	:BUS1:I2C:SDA:SOURce .....	200
	:BUS1:PARAllel:BIT<x>:SOURce .....	200
	:BUS1:PARAllel:CLOCK:EDGE .....	201
	:BUS1:PARAllel:CLOCK:SOURce .....	201
	:BUS1:PARAllel:WIDth .....	201
	:BUS1:UART:BITRate .....	202
	:BUS1:UART:DATABits .....	202
	:BUS1:UART:PARItY .....	202

	:BUS1:UART:PACKEt .....	203
	:BUS1:UART:POLARity .....	203
	:BUS1:UART:EOFPacket .....	203
	:BUS1:UART:TX:SOURce .....	204
	:BUS1:UART:RX:SOURce .....	204
	:BUS1:SPI:SCLK:POLARity .....	205
	:BUS1:SPI:SS:POLARity .....	205
	:BUS1:SPI:WORDSize .....	205
	:BUS1:SPI:BITORder .....	206
	:BUS1:SPI:SCLK:SOURce .....	206
	:BUS1:SPI:SS:SOURce.....	206
	:BUS1:SPI:MOSI:SOURce .....	207
	:BUS1:SPI:MISO:SOURce .....	207
	:BUS1:DISPlay:FORMAt .....	208
	:LISTer:DATA .....	208
	:BUS1:CAN:SOURce.....	208
	:BUS1:CAN:PROBe.....	209
	:BUS1:CAN:SAMPLEpoint.....	209
	:BUS1:CAN:BITRate .....	209
	:BUS1:LIN:BITRate .....	210
	:BUS1:LIN:IDFORmat .....	210
	:BUS1:LIN:POLARity .....	211
	:BUS1:LIN:SAMPLEpoint .....	211
	:BUS1:LIN:SOURce .....	211
	:BUS1:LIN:STANDard .....	212
<hr/>		
Mark Commands	:MARK .....	213
	:MARK:CREATE.....	213
	:MARK:DELEte.....	214
<hr/>		
Mask Commands	:MASK:STATe.....	216
	:MASK:SOURce .....	216
	:MASK:VIOLation .....	216
	:MASK:VIOLation:SAVe.....	217
	:MASK:AUTO .....	217

---

---

:MASK:AUTO:SOURce .....	218
:MASK:AUTO:UNITs .....	218
:MASK:USER:UNITs .....	218
:MASK:USER:AREa<x1>:POINt<x2> .....	219
:MASK:USER:AREa<x1>:POINt<x2>:STATe .....	219
:MASK:USER:SAVe .....	220
:MASK:USER:LOAD .....	220
:MASK:USER:CREATE .....	220
:MASK:RATio .....	221
:MASK:VIOLation:THREshold .....	221
:MASK:STOPAFter .....	221
:MASK:STOPAFter:WAVEform .....	222
:MASK:STOPAFter:TIME .....	222
:MASK:FAILACTion:STOPACQuisition .....	223
:MASK:FAILACTion:SAVEIMAge .....	223
:MASK:FAILACTion:SAVEWAVEform .....	223
:MASK:FAILACTion:HARDCopy .....	224
:MASK:FAILACTion:GONogo .....	224
:MASK:COMPLETEACTion:GONogo .....	225
:MASK:COMPLETEACTion:TESTREpeat .....	225
:MASK:COMPLETEACTion:TESTDelay .....	226
:MASK:RESults:DISPlay .....	226
:MASK:RESults:DISPlay:DETail .....	226
:MASK:RESults:CURRent:STATUS .....	227
:MASK:RESults:CURRent:WAVEform .....	227
:MASK:RESults:CURRent:VIOLation .....	227
:MASK:RESults:CURRent:TIME .....	228
:MASK:RESults:CURRent:HITSPERSEGment<x> .....	228
:MASK:RESults:TOTAL:WAVEform .....	228
:MASK:RESults:TOTAL:VIOLation .....	228
:MASK:RESults:TOTAL:TESTNUMBER .....	229
:MASK:RESults:TOTAL:FTESTNUMBER .....	229
:MASK:RESults:TOTAL:TIME .....	229
:MASK:RESults:TOTAL:HITSPERSEGment<x> .....	229

---

FRA Commands	:FRA:RUN.....	230
	:FRA:STOP .....	231
	:FRA:SOURce:INPut .....	231
	:FRA:SOURce:OUTPut .....	231
	:FRA:FREQuency:STARt .....	232
	:FRA:FREQuency:STOP .....	232
	:FRA:AWG:LOAD.....	232
	:FRA:AWG:AMPlitude.....	233
	:FRA:POINt .....	234
	:FRA:SAVe .....	234
	:FRA:RECALL.....	234
	:FRA:DATA .....	235
	:FRA:SAVETOCsv.....	235
	:FRA:AWG:AMPlitude:PROFile .....	235
	:FRA:AWG:AMPlitude:INTERPolation .....	235
	:FRA:DATA:GMARgin.....	236
	:FRA:DATA:GMARgin:FREQuency.....	236
	:FRA:DATA:PMARgin .....	237
	:FRA:DATA:PMARgin:FREQuency .....	237
	:FRA:STATE .....	237

Search	:SEARCH:COPY .....	240
Commands	:SEARCH:STATE .....	240
	:SEARCH:TOTAL .....	240
	:SEARCH:TRIGger:TYPe.....	241
	:SEARCH:TRIGger:SOURce.....	241
	:SEARCH:TRIGger:EDGE:SLOP .....	241
	:SEARCH:TRIGger:LEVel .....	242
	:SEARCH:TRIGger:HLEVel .....	243
	:SEARCH:TRIGger:LLEVel .....	243
	:SEARCH:TRIGger:PULSEWidth:POLarity .....	243
	:SEARCH:TRIGger:RUNT:POLarity .....	244
	:SEARCH:TRIGger:RISEFall:SLOP .....	244
	:SEARCH:TRIGger:PULSE:WHEn.....	245
	:SEARCH:TRIGger:PULSE:TIME.....	245

---

:SEARCH:TRIGger:RUNT:WHEn .....	246
:SEARCH:TRIGger:RUNT:TIME.....	246
:SEARCH:TRIGger:RISEFall:WHEn .....	247
:SEARCH:TRIGger:RISEFall:TIME .....	247
:SEARCH:TRIGger:BUS:TYPE .....	248
:SEARCH:TRIGger:BUS:B1:I2C:CONDition.....	248
:SEARCH:TRIGger:BUS:B1:I2C:ADDRes:MODE ...	249
:SEARCH:TRIGger:BUS:B1:I2C:ADDRes:TYPE .....	249
:SEARCH:TRIGger:BUS:B1:I2C:ADDRes:VALue ....	250
:SEARCH:TRIGger:BUS:B1:I2C:ADDRes	
:DIRection .....	251
:SEARCH:TRIGger:BUS:B1:I2C:DATA:SIZE.....	251
:SEARCH:TRIGger:BUS:B1:I2C:DATA:VALue.....	252
:SEARCH:TRIGger:BUS:B1:UART:CONDition.....	253
:SEARCH:TRIGger:BUS:B1:UART:RX:DATA:SIZE ....	253
:SEARCH:TRIGger:BUS:B1:UART:RX:DATA:VALue .	254
:SEARCH:TRIGger:BUS:B1:UART:TX:DATA:SIZE ....	256
:SEARCH:TRIGger:BUS:B1:UART:TX:DATA:VALue .	256
:SEARCH:TRIGger:BUS:B1:SPI:CONDition.....	257
:SEARCH:TRIGger:BUS:B1:SPI:DATA:SIZE.....	257
:SEARCH:TRIGger:BUS:B1:SPI:DATA:MISO:VALue	258
:SEARCH:TRIGger:BUS:B1:SPI:DATA:MOSI:VALue	259
:SEARCH:TRIGger:BUS:B1:CAN:CONDition .....	259
:SEARCH:TRIGger:BUS:B1:CAN:FRAMEtype .....	260
:SEARCH:TRIGger:BUS:B1:CAN:IDentifier:MODE .	261
:SEARCH:TRIGger:BUS:B1:CAN:IDentifier:VALue ..	261
:SEARCH:TRIGger:BUS:B1:CAN:IDentifier	
:DIRection .....	262
:SEARCH:TRIGger:BUS:B1:CAN:DATA:QUALifier...	263
:SEARCH:TRIGger:BUS:B1:CAN:DATA:SIZE .....	264
:SEARCH:TRIGger:BUS:B1:CAN:DATA:VALue .....	264
:SEARCH:TRIGger:BUS:B1:LIN:CONDition .....	265
:SEARCH:TRIGger:BUS:B1:LIN:DATA:QUALifier ....	266
:SEARCH:TRIGger:BUS:B1:LIN:DATA:SIZE .....	267
:SEARCH:TRIGger:BUS:B1:LIN:DATA:VALue .....	267

---

:SEARCH:TRIGger:BUS:B1:LIN:ERRTYPE .....	268
:SEARCH:TRIGger:BUS:B1:LIN:IDentifier:VALue ...	268
:SEARCH:FFTPeak:METhod .....	269
:SEARCH:FFTPeak:METhod:MPeak .....	270
:SEARCH:FFTPeak:SINFo .....	270
:SEARCH:FFTPeak:LIST .....	271
:SEARCH:FFTPeak:SOURce .....	271
:SEARCH:TRIGger:LOGic:INPut:CLOCK:SOURce ..	272
:SEARCH:TRIGger:LOGic:INPut:CLOCK:EDGe .....	272
:SEARCH:TRIGger:LOGic:FUNcTion .....	273
:SEARCH:TRIGger:LOGic:PATtern .....	273
:SEARCH:TRIGger:LOGic:PATtern:INPut:D<x> .....	274
:SEARCH:TRIGger:LOGic:PATtern:DELtAtime .....	274
:SEARCH:TRIGger:LOGic:PATtern:WHEn .....	275
:SEARCH:TRIGger:BUS:TYPe .....	275

Label Commands	:CHANnel<X>:LABel .....	277
	:CHANnel<X>:LABel:DISPlay .....	278
	:REF<X>:LABel .....	278
	:REF<X>:LABel:DISPlay .....	279
	:BUS1:LABel .....	280
	:BUS1:LABel:DISPlay .....	280
	:D<x>:LABel .....	281
	:D<x>:LABel:DISPlay .....	282
	:DIGital:ANALog:A<x>:LABel .....	282
	:DIGital:ANALog:A<x>:LABel:DISPlay .....	283
	:SET<X>:LABel .....	284

Segment	:SEGMENTS:STATE .....	285
Commands	:SEGMENTS:CURRent .....	286
	:SEGMENTS:TOTALnum .....	286
	:SEGMENTS:TIME .....	287
	:SEGMENTS:DISPALL .....	287
	:SEGMENTS:MEASURE:MODE .....	288
	:SEGMENTS:MEASURE:PLOT:SOURce .....	288

---

	:SEGMENTS:MEASURE:PLOT:DIVIDE .....	289
	:SEGMENTS:MEASURE:PLOT:SELECT .....	289
	:SEGMENTS:MEASURE:PLOT:RESULTS .....	289
	:SEGMENTS:MEASURE:TABLE:SOURCE .....	290
	:SEGMENTS:MEASURE:TABLE:SELECT .....	291
	:SEGMENTS:MEASURE:TABLE:LIST .....	291
	:SEGMENTS:MEASURE:TABLE:SAVE .....	292
	:SEGMENTS:SAVE .....	292
	:SEGMENTS:SAVE:SOURCE .....	292
	:SEGMENTS:SAVE:SELECT:START .....	293
	:SEGMENTS:SAVE:SELECT:END .....	293
	:SEGMENTS:MASK:STATE .....	294
	:SEGMENTS:MASK:SOURCE .....	294
	:SEGMENTS:MASK:CREATE .....	294
	:SEGMENTS:MASK:UNITS .....	295

---

DVM Commands	:DVM:STATE .....	296
	:DVM:SOURCE .....	296
	:DVM:MODE .....	297
	:DVM:VALUE .....	297

---

Go-NoGo	:GONOGO:CLEAR .....	298
Commands	:GONOGO:EXECUTE .....	298
	:GONOGO:FUNCTION .....	299
	:GONOGO:NGCOUNT .....	299
	:GONOGO:NGDEFINE .....	299
	:GONOGO:SOURCE .....	300
	:GONOGO:VIOLATION .....	300
	:GONOGO:SCRIPT .....	300
	:TEMPLATE:MODE .....	301
	:TEMPLATE:MAXIMUM .....	301
	:TEMPLATE:MINIMUM .....	301
	:TEMPLATE:POSITION:MAXIMUM .....	302
	:TEMPLATE:POSITION:MINIMUM .....	302
	:TEMPLATE:SAVE:MAXIMUM .....	302

---

---

	:TEMPLate:SAVe:MINimum .....	303
	:TEMPLate:TOLerance.....	303
	:TEMPLate:SAVe:AUTo .....	303

---

AWG Commands	:AWG:UTIL.....	305
	:AWG:UTIL:AMPcpl .....	305
	:AWG:UTIL:FREQcpl.....	306
	:AWG:UTIL:FREQcpl:OFFSet .....	306
	:AWG:UTIL:FREQcpl:RATio.....	307
	:AWG:UTIL:TRACking.....	307
	:AWG<x>:AMPlitude.....	307
	:AWG<x>:FREQUency.....	308
	:AWG<x>:FUNCTion .....	308
	:AWG<x>:OFFSet .....	309
	:AWG<x>:OUTPut:LOAd:IMPEDance .....	309
	:AWG<x>:OUTPut:STATE .....	310
	:AWG<x>:PHAsE.....	310
	:AWG<x>:PULSe:DUTYcycle .....	310
	:AWG<x>:RAMP:SYMmetry .....	311
	:AWG<x>:MODulation:STATE .....	311
	:AWG<x>:MODulation:TYPe.....	311
	:AWG<x>:MODulation:AM:DEPth.....	312
	:AWG<x>:MODulation:AM:FREQ.....	312
	:AWG<x>:MODulation:AM:SHApe.....	312
	:AWG<x>:MODulation:AM:PHAsE .....	313
	:AWG<x>:MODulation:AM:DUTYcycle .....	313
	:AWG<x>:MODulation:AM:SYMmetry .....	313
	:AWG<x>:MODulation:AM:RATE .....	314
	:AWG<x>:MODulation:FM:DEV .....	314
	:AWG<x>:MODulation:FM:FREQ.....	315
	:AWG<x>:MODulation:FM:SHApe .....	315
	:AWG<x>:MODulation:FM:PHAsE .....	316
	:AWG<x>:MODulation:FM:DUTYcycle.....	316
	:AWG<x>:MODulation:FM:SYMmetry .....	316
	:AWG<x>:MODulation:FM:RATE.....	317

---



---

	:AWG<x>:MODulation:FSK:FREQ .....	317
	:AWG<x>:MODulation:FSK:RATE .....	318
	:AWG<x>:SWEp:STATE .....	318
	:AWG<x>:SWEp:TYPe .....	318
	:AWG<x>:SWEp:START .....	319
	:AWG<x>:SWEp:STOP .....	319
	:AWG<x>:SWEp:TIME .....	319
	:AWG<x>:SWEp:SPAN .....	320
	:AWG<x>:SWEp:CENTer .....	320
	:AWG<x>:ARBitrary:EDIT:NUMPoint .....	320
	:AWG<x>:ARBitrary:EDIT:FUNCTion.....	321
	:AWG<x>:ARBitrary:SAVe:WAVEform .....	321
	:AWG<x>:ARBitrary:LOAd:WAVEform .....	322
	:AWG<x>:ARBitrary:EDIT:COpy.....	322
	:AWG<x>:ARBitrary:EDIT:CLear.....	323
	:AWG<x>:ARBitrary:EDIT:LINE .....	323
	:AWG<x>:ARBitrary:EDIT:SCALE.....	324
	:AWG<x>:ARBitrary:EDIT:POINT.....	324
	:AWG<x>:ARBitrary:EDIT:POINT:ADD .....	324
	:AWG<x>:ARBitrary:EDIT:POINT:DELEte.....	325

---

Data Logging	:DATALOG:STATE .....	326
Commands	:DATALOG:SOURce .....	326
	:DATALOG:SAVe .....	327
	:DATALOG:INTerval .....	327
	:DATALOG:DURation .....	328

---

Remote Disk	:REMOTEDisk:IPADDRESS .....	329
Commands	:REMOTEDisk:PATHName .....	329
	:REMOTEDisk:USERName .....	329
	:REMOTEDisk:PASSWord .....	330
	:REMOTEDisk:MOUNT.....	330
	:REMOTEDisk:AUTOMount.....	331

---

Spectrum	:SA:STATE .....	333
----------	-----------------	-----

---

---

Analyzer	:SA:LIST.....	333	
Commands	:SA:MEMory.....	333	
	:SA:MEMory:SOURce.....	335	
	:SA:SOURce.....	336	
	:SA<x>:SPECTRUMTrace.....	336	
	:SA<x>:NORMal.....	337	
	:SA<x>:MAXHold.....	337	
	:SA<x>:MINHold.....	337	
	:SA<x>:AVErage.....	338	
	:SA<x>:AVErage:NUMAVg.....	338	
	:SA<x>:DETECTIonmethod:MODE.....	339	
	:SA<x>:DETECTIonmethod:MAXHold.....	339	
	:SA<x>:DETECTIonmethod:MINHold.....	340	
	:SA<x>:DETECTIonmethod:NORMal.....	340	
	:SA<x>:DETECTIonmethod:AVErage.....	341	
	:SA<x>:FREQuency.....	341	
	:SA<x>:SPAN.....	342	
	:SA<x>:START.....	342	
	:SA<x>:STOP.....	342	
	:SA<x>:RBW:MODE.....	343	
	:SA<x>:RBW.....	343	
	:SA<x>:SPANRbwratio.....	344	
	:SA<x>:WINDow.....	344	
	:SA<x>:UNIts.....	345	
	:SA<x>:SCALE.....	345	
	:SA:POSition.....	346	
	:SA<x>:INPut.....	346	
	:SA:SPECTRogram:NUMSLICES?.....	347	
	:SA<x>:SPECTRogram:SLICESElect.....	347	
	:SA<x>:SPECTRogram:SLICETIME?.....	347	
	:SA:SPECTRogram:STATe.....	348	
	<hr/>		
	Power Analysis	:POWer:TYPe.....	355
Commands	:POWer:CURREnt:SOURce.....	355	
	:POWer:VOLTage:SOURce.....	356	

---

---

:POWER:HARMonics:STANdard .....	356
:POWER:HARMonics:NR_HARMonics .....	356
:POWER:HARMonics:SOURce .....	357
:POWER:HARMonics:FREQRef .....	357
:POWER:HARMonics:FREQRef:FIXEDFREQValue...358	
:POWER:HARMonics:DISPlay:SElect .....	358
:POWER:HARMonics:DISPlay:TYPe.....	359
:POWER:HARMonics:IEC:LINEFREQuency.....	359
:POWER:HARMonics:IEC:OBSPERiod.....	359
:POWER:HARMonics:IEC:CLAss.....	360
:POWER:HARMonics:IEC:POWERFACTOR .....	360
:POWER:HARMonics:IEC:FUNDamental .....	360
:POWER:HARMonics:IEC:INPUTPOWER .....	361
:POWER:HARMonics:IEC:FILter .....	361
:POWER:HARMonics:IEC:GROUPing.....	361
:POWER:HARMonics:RESults:HAR<1-400>	
:REQuency? .....	362
:POWER:HARMonics:RESults:HAR<1-40>:ECMAX?362	
:POWER:HARMonics:RESults:HAR<1-40>:LIMit? ...362	
:POWER:HARMonics:RESults:HAR<1-400>:HASE? .363	
:POWER:HARMonics:RESults:HAR<1-400>:RMS	
:ABSolute? .....	363
:POWER:HARMonics:RESults:HAR<1-400>:RMS	
:PERCent? .....	363
:POWER:HARMonics:RESults:HAR<1-40>:TEST	
:IEC:CLASSALIMit? .....	364
:POWER:HARMonics:RESults:HAR<1-40>:TEST	
:IEC:NORMAL? .....	364
:POWER:HARMonics:RESults:HAR<1-40>:TEST	
:IEC:POHCLIMit? .....	364
:POWER:HARMonics:RESults:IEC:FUNDamental?..365	
:POWER:HARMonics:RESults:IEC	
:HARM3ALTernate? .....	365
:POWER:HARMonics:RESults:IEC	
:HARM5ALTernate? .....	366
:POWER:HARMonics:RESults:IEC:POHC? .....	366

---

:POWer:HARMonics:RESults:IEC:POWer?	366
:POWer:HARMonics:RESults:IEC:POWERFactor?	366
:POWer:HARMonics:RESults:PASSFail?	367
:POWer:HARMonics:RESults:RMS?	367
:POWer:HARMonics:RESults:THDF?	367
:POWer:HARMonics:RESults:THDR?	368
:POWer:HARMonics:RESults:SAVe	368
:POWer:GATing	368
:POWer:QUALity:DISPlay	368
:POWer:QUALity:FREQREference	369
:POWer:QUALity:VMAX?	369
:POWer:QUALity:VMIN?	370
:POWer:QUALity:IMAX?	370
:POWer:QUALity:IMIN?	370
:POWer:QUALity:DCVOLTage?	370
:POWer:QUALity:DCCURRent?	371
:POWer:QUALity:VCRESTfactor?	371
:POWer:QUALity:ICRESTfactor?	371
:POWer:QUALity:IMPedance?	371
:POWer:QUALity:RESIstance?	372
:POWer:QUALity:REACtance?	372
:POWer:QUALity:APPpwr?	372
:POWer:QUALity:FREQUency?	372
:POWer:QUALity:ICRESTfactor?	373
:POWer:QUALity:IRMS?	373
:POWer:QUALity:PHASEangle?	373
:POWer:QUALity:POWERFACTor?	373
:POWer:QUALity:REACTpwr?	374
:POWer:QUALity:TRUEpwr?	374
:POWer:QUALity:VRMS?	374
:POWer:RIPPLE	374
:POWer:RIPPLE:RESults:AMPlitude?	375
:POWer:RIPPLE:RESults:REALamplitude?	375
:POWer:RIPPLE:SOURce	375
:POWer:INRUshcurrent:RESults:FIRStpeak?	375

---

---

:POWer:INRUshcurrent:RESUltS:SECOndpeak? .....	376
:POWer:MODUlation:SOUrce .....	376
:POWer:MODUlation:TYPe.....	376
:POWer:REFLevel:PERCent:MID .....	377
:POWer:SOA:AXESTYPE .....	377
:POWer:SOA:LINear:XMAX.....	377
:POWer:SOA:LINear:XMIN .....	378
:POWer:SOA:LINear:YMAX.....	378
:POWer:SOA:LINear:YMIN .....	378
:POWer:SOA:LOG:XMAX .....	379
:POWer:SOA:LOG:XMIN .....	379
:POWer:SOA:LOG:YMAX .....	379
:POWer:SOA:LOG:YMIN.....	380
:POWer:SOA:MASK:TYPe.....	380
:POWer:SOA:MASK:NUMPt .....	380
:POWer:SOA:MASK:POINt<x> .....	381
:POWer:SOA:MASK:MAXVoltage.....	381
:POWer:SOA:MASK:MAXCURRent.....	381
:POWer:SOA:MASK:MAXPower.....	382
:POWer:SOA:MASK:STOPOnviol .....	382
:POWer:SOA:RESUltS:FAILures? .....	382
:POWer:SOA:RESUltS:NUMACq? .....	382
:POWer:SOA:MASK:RESUltS:STATe? .....	383
:POWer:SWLoss:CONDCALCtype .....	383
:POWer:SWLoss:DISPlay.....	383
:POWer:SWLoss:GATing .....	384
:POWer:SWLoss:RDson .....	384
:POWer:SWLoss:VCEsat.....	384
:POWer:SWLoss:REFLevel:PERCent:MID.....	385
:POWer:SWLoss:REFLevel:PERCent:CURRent .....	385
:POWer:SWLoss:REFLevel:PERCent:VOLTage .....	385
:POWer:SWLoss:CONDUction:ENERgy:MAX? .....	385
:POWer:SWLoss:CONDUction:ENERgy:MEAN? .....	386
:POWer:SWLoss:CONDUction:ENERgy:MIN? .....	386
:POWer:SWLoss:CONDUction:POWer:MAX? .....	386

---

:POWer:SWLoss:CONDUction:POWer:MEAN?	.....	386
:POWer:SWLoss:CONDUction:POWer:MIN?	.....	386
:POWer:SWLoss:TOFF:ENERgy:MAX?	.....	387
:POWer:SWLoss:TOFF:ENERgy:MEAN?	.....	387
:POWer:SWLoss:TOFF:ENERgy:MIN?	.....	387
:POWer:SWLoss:TOFF:POWer:MAX?	.....	387
:POWer:SWLoss:TOFF:POWer:MEAN?	.....	387
:POWer:SWLoss:TOFF:POWer:MIN?	.....	388
:POWer:SWLoss:TON:ENERgy:MAX?	.....	388
:POWer:SWLoss:TON:ENERgy:MEAN?	.....	388
:POWer:SWLoss:TON:ENERgy:MIN?	.....	388
:POWer:SWLoss:TON:POWer:MAX?	.....	388
:POWer:SWLoss:TON:POWer:MEAN?	.....	389
:POWer:SWLoss:TON:POWer:MIN?	.....	389
:POWer:SWLoss:TOTAl:ENERgy:MAX?	.....	389
:POWer:SWLoss:TOTAl:ENERgy:MEAN?	.....	389
:POWer:SWLoss:TOTAl:ENERgy:MIN?	.....	389
:POWer:SWLoss:TOTAl:POWer:MAX?	.....	390
:POWer:SWLoss:TOTAl:POWer:MEAN?	.....	390
:POWer:SWLoss:TOTAl:POWer:MIN?	.....	390
:POWer:TRANSient:DURation	.....	390
:POWer:TRANSient:OVERShoot	.....	390
:POWer:TRANSient:STEADYVout	.....	391
:POWer:TRANSient:LOWCurrent	.....	391
:POWer:TRANSient:HIGHCurrent	.....	391
:POWer:TRANSient:RESults:VALue?	.....	392
:POWer:EFFiciency:STATIstics:MODE	.....	392
:POWer:EFFiciency:STATIstics:WEIghting	.....	392
:POWer:EFFiciency:STATIstics	.....	392
:POWer:EFFiciency:INPUTPOWer:VALue?	.....	393
:POWer:EFFiciency:INPUTPOWer:MEAN?	.....	393
:POWer:EFFiciency:INPUTPOWer:MINImum?	.....	393
:POWer:EFFiciency:INPUTPOWer:MAXmum?	.....	393
:POWer:EFFiciency:INPUTPOWer:STDdev?	.....	393
:POWer:EFFiciency:OUTPUTPOWer:VALue?	.....	394

---

:POWer:EFFiciency:OUTPUTPOWer:MEAN?	394
:POWer:EFFiciency:OUTPUTPOWer:MINImum?	394
:POWer:EFFiciency:OUTPUTPOWer:MAXmum?	394
:POWer:EFFiciency:OUTPUTPOWer:STDdev?	394
:POWer:EFFiciency:RESults:VALue?	395
:POWer:EFFiciency:RESults:MEAN?	395
:POWer:EFFiciency:RESults:MINImum?	395
:POWer:EFFiciency:RESults:MAXmum?	395
:POWer:EFFiciency:RESults:STDdev?	395
:POWer:BHCurve:WINDings	396
:POWer:BHCurve:CROSECArea	396
:POWer:BHCurve:MAGNLength	396
:POWer:BHCurve:B:POSition	397
:POWer:BHCurve:B:SCAle	397
:POWer:BHCurve:H:POSition	397
:POWer:BHCurve:H:SCAle	397
:POWer:TONOff:TEST	398
:POWer:TONOff:SETup:DURation	398
:POWer:TONOff:SETup:APPLY	398
:POWer:TONOff:SETup:SAVESetup	399
:POWer:TONOff:DISPlay	399
:POWer:SHOrt:DISPlay	399
:POWer:SHOrt:APPLY	400
:POWer:OCP:DISPlay	400
:POWer:OCP:APPLY	400
:POWer:PSRR:RUN	400
:POWer:PSRR:STOP	401
:POWer:PSRR:SOURce:INPut	401
:POWer:PSRR:SOURce:OUTPut	401
:POWer:PSRR:FREQuency:START	401
:POWer:PSRR:FREQuency:STOP	402
:POWer:PSRR:AWG:LOAd	402
:POWer:PSRR:AWG:AMPlitude	402
:POWer:PSRR:POINt	403
:POWer:PSRR:SAVE	404

---

:POWer:PSRR:RECALL.....	404
:POWer:PSRR:DATA .....	404
:POWer:PSRR:SAVETOCsv.....	405
:POWer:PSRR:AWG:AMPlitude:PROFile .....	405
:POWer:PSRR:AWG:AMPlitude:INTERPolation .....	405
:POWer:PSRR:DATA:GMARgin .....	406
:POWer:PSRR:DATA:GMARgin:FREQuency.....	406
:POWer:PSRR:DATA:PMARgin .....	406
:POWer:PSRR:DATA:PMARgin:FREQuency .....	407
:POWer:CLR:RUN .....	407
:POWer:CLR:STOP .....	407
:POWer:CLR:SOURce:INPut .....	407
:POWer:CLR:SOURce:OUTPut .....	408
:POWer:CLR:FREQuency:STARt.....	408
:POWer:CLR:FREQuency:STOP .....	408
:POWer:CLR:AWG:LOAD.....	409
:POWer:CLR:AWG:AMPlitude .....	409
:POWer:CLR:POINT .....	410
:POWer:CLR:SAVE .....	410
:POWer:CLR:RECALL.....	410
:POWer:CLR:DATA .....	411
:POWer:CLR:SAVETOCsv .....	411
:POWer:CLR:AWG:AMPlitude:PROFile.....	411
:POWer:CLR:AWG:AMPlitude:INTERPolation.....	412
:POWer:CLResponse:DATA:GMARgin .....	412
:POWer:CLResponse:DATA:GMARgin:FREQuency	413
:POWer:CLResponse:DATA:PMARgin .....	413
:POWer:PSRR:DATA:PMARgin:FREQuency .....	413

USB Delay Command	:USBDelay .....	414
-------------------	-----------------	-----

Digital Commands	:D<x>:DISPlay .....	415
	:D<x>:POSition .....	415
	:DISPlay:DIGital:HEIght .....	416
	:DIGital:GROUP<x>:THReshold .....	416



:DIGital:ANALog:A<x>:DISPlay .....	417
:DIGital:ANALog:A<x>:RATio .....	417
:DIGital:ANALog:A<x>:POSition .....	418
:D<x>:MEMory .....	418
:DIGital:MEMory .....	420

---

# 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 .....	44
Acquisition Commands .....	50
Autoscale Commands .....	59
Vertical Commands .....	60
Math Commands .....	66
Cursor Commands .....	77
Display Commands .....	90
Hardcopy Commands.....	96
Measure Commands .....	99
Measurement Commands .....	127
Reference Commands .....	136
Run Command .....	139
Stop Command .....	139
Single Command.....	139
Force Command.....	140
Timebase Commands.....	141
Trigger Commands.....	144
System Commands .....	186
Save/Recall Commands.....	187
Ethernet Commands .....	192
Time Command .....	196
Bus Decode Commands .....	197
Mark Commands.....	213
Mask Commands .....	215

FRA Commands .....	230
Search Commands .....	238
Label Commands .....	277
Segment Commands .....	285
DVM Commands.....	296
Go_NoGo Commands.....	298
AWG Commands.....	304
Data Logging Commands.....	326
Remote Disk Commands.....	329
Spectrum Analyzer Commands .....	332
Power Analysis Commands .....	349
USB Delay Command.....	414
Digital Commands .....	415

## Common Commands

*IDN?	44
*LRN?	44
*SAV	45
*RCL	45
*RST	46
*CLS	46
*ESE	46
*ESR	47
*OPC	47
*SRE	48
*STB	49

### \*IDN? → Query

**Description** Returns the manufacturer, model, serial number and version number of the unit.

**Syntax** \*IDN?

**Example** \*IDN?  
GW-INSTEK,GDS-3654A,PXXXXXX,V1.00

### \*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-3000A (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.

**\*OPC**

Set →

→ 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 .....	50
:ACQuire:MODE .....	51
:ACQuire<X>:MEMory? .....	51
:ACQuire:FILTer:SOURce .....	53
:ACQuire:FILTer .....	54
:ACQuire:FILTer:FREQuency.....	54
:ACQuire:FILTer:FREQuency:UPPER .....	55
:ACQuire:FILTer:FREQuency:LOWER .....	55
:ACQuire:FILTer:TYPe .....	55
:ACQuire:FILTer:TRACking.....	56
:ACQuire<X>:STATe? .....	56
:ACQuire:INTERpolation .....	57
:ACQuire:RECOrdlength.....	57
:HEADer .....	57
: ACQuire:SAMPLerate? .....	58

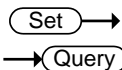
### :ACQuire:AVERage

Set →

→ Query

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, 512
 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.



**:ACquire:MODE**

Description	Selects or returns the acquisition mode.	
Syntax	:ACquire:MODE {SAMPLE   PDETECT   HIRes   AVERage   ?}	
Related Commands	:ACquire:AVERage	
Parameter	SAMPLE	Sample mode sampling
	PDETECT	Peak detect sampling
	HIRes	High resolution sampling
	AVERage	Average sampling mode
Example	:ACquire:MODE PDETECT Sets the sampling mode to peak detection.	

**:ACquire<X>:MEMory?**

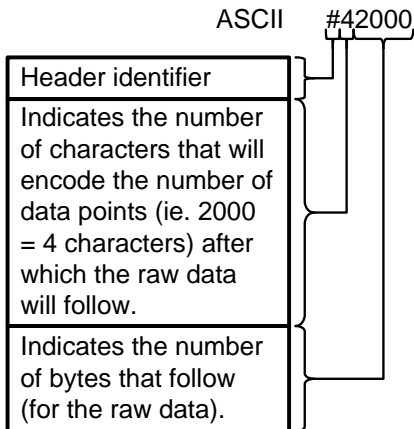


Description	Returns the data in acquisition memory for the selected channel as a header + raw data.	
Syntax	:ACquire<X>:MEMory?	
Related Commands	:ACquire:RECOrdlength :HEADer	
Parameter	<X>	Channel number (1 to 4)
Return parameter		Returns acquisition settings followed by raw waveform block data.
	<string>	<string> Returns the acquisition settings for the selected channel.

Format:  
 parameter(1),setting(1);parameter(2),s  
 etting(2)...parameter(n),setting(n);Wa  
 veform Data;

<waveform  
 block 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.....
```

Set →

→ Query

:ACquire:FILTer:SOURce

Description      Sets or 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.

**:ACquire:FILTer:FREQuency:UPPER** 


Description	Sets or returns the filter upper frequency.	
Syntax	:ACquire:FILTer:FREQuency:UPPER {DEfault} :ACquire:FILTer:FREQuency:UPPER <NRf> :ACquire:FILTer:FREQuency:UPPER?	
Parameter	DEfault	Sets the frequency to default.
	<NRf>	Sets the frequency to user. (Range:1Hz~500MHz)
Example	:ACquire:FILTer:FREQuency:UPPER 4.95e+07 :ACquire:FILTer:FREQuency:UPPER? 4.950000e+07	

**:ACquire:FILTer:FREQuency:LOWER** 


Description	Sets or returns the filter lower frequency.	
Syntax	:ACquire:FILTer:FREQuency:LOWER {DEfault} :ACquire:FILTer:FREQuency:LOWER <NRf> :ACquire:FILTer:FREQuency:LOWER?	
Parameter	DEfault	Sets the frequency to default.
	<NRf>	Sets the frequency to user. (Range:1Hz~500MHz)
Example	:ACquire:FILTer:FREQuency:LOWER 1.25e+07 :ACquire:FILTer:FREQuency:LOWER? 1.250000e+07	

**:ACquire:FILTer:TYPE** 


Description	Sets or returns the filter type.	
-------------	----------------------------------	--

Syntax	:ACquire:FILTer:TRACking {LOWPass   HIGHPass  BANDPass}	
	:ACquire:FILTer:TYPe?	
Parameter	LOWPass	Lowpass Type.
	HIGHPass	Highpass Type.
	BANDPass	bandpass Type.
Example	:ACquire:FILTer:TYPe? >LOWPass Returns low pass type as present filter type	

:ACquire:FILTer:TRACking (Set) →  
→ (Query)

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.	





Note

If the oscilloscope changes the acquisition status from STOP to RUN, the status will be reset as zero.

Set →

→ Query

**:ACQUIRE:INTERpolation**

Description	Selects or returns the interpolation mode.	
Syntax	:ACQUIRE:INTERpolation {ET   SINC   ?}	
Parameter/Return parameter	ET	Equivalent Time interpolation. The GDS-3000A doesn't support ET.
	SINC	Sets to SIN(X)/X interpolation
Example	:ACQUIRE:INTERpolation? >SINC Returns SINC as the interpolation mode.	

Set →

→ Query

**:ACQUIRE:RECORDlength**

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.	

Set →

→ Query

**:HEADER**

Description	Configures whether the returned data of the :ACQUIRE:MEM query will contain header information or not. It is set to ON by default.	
Syntax	:HEADER {OFF   ON   ?}	
Related Commands	:ACQUIRE<X>:MEMORY?	
Parameter	ON	Add header information.

OFF Don't add header information.

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

Example :HEADer ON

**:ACQuire:SAMPlerate?**

→ Query

Description Querys the value of sample rate.

Syntax : ACQuire:SAMPlerate?

Example : ACQuire:SAMPlerate?

1.00000E+09

## Autoscale Commands

:AUTOSet .....	59
:AUTORSET:MODE .....	59

### :AUTOSet (Set) →

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

**Syntax**            :AUTOSet

(Set) →

### :AUTORSET:MODE → (Query)

**Description**      Sets the Autoset mode or queries its state.

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

**Related Commands**      :AUTOSet

<b>Parameter/Return parameter</b>	FITScreen	Fit Screen mode
	ACPriority	AC priority mode

**Example**            :AUTORSET?  
FITSCREEN

## Vertical Commands

:CHANnel<X>:BWLimit.....	60
:CHANnel<X>:COUPling .....	61
:CHANnel<X>:DESKew .....	61
:CHANnel<X>:DISPlay .....	61
:CHANnel<X>:EXPand .....	62
:CHANnel<X>:IMPedance?.....	62
:CHANnel<X>:INVert .....	63
:CHANnel<X>:POSition .....	63
:CHANnel<X>:PROBe:RATio .....	64
:CHANnel<X>:PROBe:TYPe.....	64
:CHANnel<X>:SCALE .....	65

: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.	

Set →  
→ Query

**: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.	

Set →  
→ Query

**: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.	

Set →  
→ Query

**: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

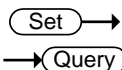
Description Sets or returns the impedance of the oscilloscope.

Syntax :CHANnel<X>:IMPedance (50 | 1M)  
:CHANnel<X>:IMPedance?

Parameter	50   1M	50Ω or 1MΩ input impedance
	<x>	Channel
	1/2/3/4	CH1/2/3/4

Return parameter <NR3> Returns the impedance value.

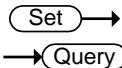
Example :CHANnel1:IMPedance?  
 1.000000E+06  
 The impedance is 1M ohms.




**: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	The vertical position will only be set to closest allowed value. The position level range depends on the vertical scale.  The scale must first be set before the position can be set.	
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 →

:CHANnel<X>:PROBe:RATio

→ Query

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 →

:CHANnel<X>:PROBe:TYPE

→ Query

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

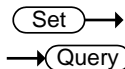
**:CHANnel<X>:SCALE**

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 1mV 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 .....	66
:MATH:TYPe .....	67
:MATH:DUAL:SOURce<X> .....	69
:MATH:DUAL:OPERator .....	70
:MATH:DUAL:POSition .....	70
:MATH:DUAL:SCALe .....	70
:MATH:FFT:SOURce .....	71
:MATH:FFT:MAG .....	71
:MATH:FFT:WINDow .....	72
:MATH:FFT:POSition .....	72
:MATH:FFT:SCALe .....	72
:MATH:FFT:HORizontal:SCALe .....	73
:MATH:FFT:HORizontal:POSition .....	73
:MATH:DEFine .....	73
:MATHVAR? .....	75
:MATHVAR:VAR<X> .....	75
:MATH:ADVanced:POSition .....	75
:MATH:ADVanced:SCALe .....	76

### :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.	

→ Set →  
→ Query

**: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	<table border="1"> <tr> <td>DUAL</td> <td>Dual channel operations</td> </tr> <tr> <td>ADVanced</td> <td>Advanced math operations</td> </tr> <tr> <td>FFT</td> <td>FFT operations</td> </tr> </table>	DUAL	Dual channel operations	ADVanced	Advanced math operations	FFT	FFT operations
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.						

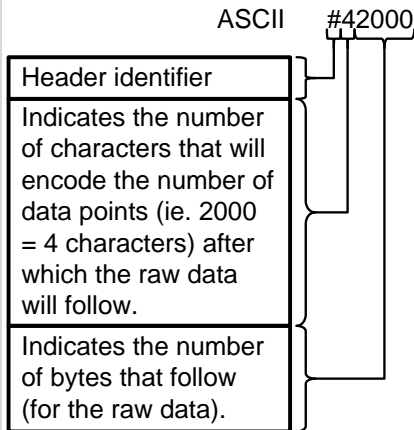
→ Query

**:MATH:MEMory?**

---

Description	Returns raw data of MATH, return data containing header with raw data				
Syntax	:MATH:MEMory?				
Related Commands	:HEADer				
Parameter	<X> Channel number (1 to 4)				
Return parameter	<table border="1"> <tr> <td></td> <td>Returns acquisition settings followed by raw waveform block data.</td> </tr> <tr> <td>&lt;string&gt;</td> <td>                     &lt;string&gt;                      Returns the acquisition settings for the selected channel.                       Format:                      parameter(1),setting(1);parameter(2),setting(2)...parameter(n),setting(n);Waveform Data;                 </td> </tr> </table>		Returns acquisition settings followed by raw waveform block data.	<string>	<string> Returns the acquisition settings for the selected channel.  Format: parameter(1),setting(1);parameter(2),setting(2)...parameter(n),setting(n);Waveform Data;
	Returns acquisition settings followed by raw waveform block data.				
<string>	<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> <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

```
:MATH:MEMory?
Format,2.0E;Memory Length,10000;IntpDistance,0;
Trigger Address,4999;Trigger Level,1.160E+01;
Source,Math;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.....
```

Set →

→ Query

**: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	<table border="0"> <tr> <td>&lt;X&gt;</td> <td>Source number 1 or 2</td> </tr> <tr> <td>CH1~4</td> <td>Channel 1 to 4</td> </tr> <tr> <td>REF1~4</td> <td>Reference waveforms 1 to 4</td> </tr> </table>	<X>	Source number 1 or 2	CH1~4	Channel 1 to 4	REF1~4	Reference waveforms 1 to 4
<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 Set →  
→ Query

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 Set →  
→ Query

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 Set →  
→ Query

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.	

Set →  
 → Query

**: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.	

Set →  
 → Query

**: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** 

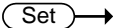
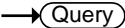
  


Description	Sets the windowing filter used for the FFT function.	
Syntax	:MATH:FFT:WINDow {RECTangular HAMming HANning BLAckman ?}	
Parameter	RECTangular	Rectangular window
	HAMming	Hamming window
	HANning	Hanning window
	BLAckman	Blackman window
Return parameter	Returns the FFT window.	
Example	:MATH:FFT:WINDow HAMming Sets the FFT window filter to hamming.	

**: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.	

**: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?**

→ Query

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 →

**:MATHVAR:VAR<X>**

→ Query

Description	Sets or returns the VAR1 or VAR2 variables.
Syntax	MATHVAR:VAR<x> {<NRf>   ?}
Related Commands	MATH:DEFine
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 →

**:MATH:ADVanced:POSition**

→ Query

Description	Sets the vertical position of the advanced math result, expressed in unit/div.
Syntax	MATH:ADVanced:POSition { <NRf>   ? }
Parameter	<NRf> Vertical position: -12e+0 to +12e+0 (12 units/division to +12 units/division.)

Return parameter <NR3> Returns the vertical position.

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

Set →

→ Query

**: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 .....	78
:CURSor:SOURce.....	78
:CURSor:HUNI .....	79
:CURSor:HUSE .....	79
:CURSor:VUNI .....	80
:CURSor:VUSE .....	80
:CURSor:DDT .....	80
:CURSor:H1Position .....	81
:CURSor:H2Position .....	81
:CURSor:HDELta .....	82
:CURSor:V1Position.....	82
:CURSor:V2Position.....	82
:CURSor:VDELta .....	83
:CURSor:XY:RECTangular:X:POSition<X> .....	83
:CURSor:XY:RECTangular:X:DELta .....	83
:CURSor:XY:RECTangular:Y:POSition<X> .....	84
:CURSor:XY:RECTangular:Y:DELta .....	84
:CURSor:XY:POLar:RADIUS:POSition<X>.....	84
:CURSor:XY:POLar:RADIUS:DELta .....	85
:CURSor:XY:POLar:THETA:POSition<X> .....	85
:CURSor:XY:POLar:THETA:DELta .....	85
:CURSor:XY:PRODuct:POSition<X> .....	86
:CURSor:XY:PRODuct:DELta.....	86
:CURSor:XY:RATio:POSition<X> .....	86
:CURSor:XY:RATio:DELta .....	87
:CURSor:HTRACking .....	87
:CURSor:VTRACking .....	87
:CURSor:MARK .....	88
:CURSor:SA:SOURce .....	88

---

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 | ? }

<b>Parameter</b>	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 | LOGic | BUS1 | NORMal | MAXHold | MINHold | AVErage | ? }

<b>Parameter</b>	CH1~CH4	Channel 1 to 4
	REF1~4	Reference waveform 1 to 4
	MATH	Math source
	LOGic	Logic source
	BUS1	Bus source
	NORMal ~ AVErage	Source in SA mode. Can be set only in SA mode


**Return parameter** Returns the cursor source.

**Example** :CURSor:SOURce CH1  
Turns the cursor source as channel 1.


:CURSor:HUNI (Set) →  
→ (Query)

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 (Set) →

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.	

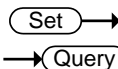
		(Set) →
		→ (Query)
<b>:CURSor:VUNI</b>		
Description	Sets or queries the units for the vertical bar cursors.	
Syntax	:CURSor:VUNI {BASE   PERcent   ?}	
Related Commands	:CURSor:MODE	
Parameter	BASE	Sets the vertical cursor units the same as the scope units (V or A).
	PERcent	Sets the displayed units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

		(Set) →
<b>:CURSor:VUSE</b>		
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)
<b>:CURSor:DDT</b>		
Description	Returns the deltaY/deltaT (dy/dT) readout. This function is only supported if the source channels are CH1~4, Ref1~4 or Math.	

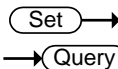


Syntax	:CURSor:DDT{?}
Related Commands	:CURSor:MODE
Return Parameter	<NR3> Returns the readout in <NR3> format.
Example	:CURSor:DDT? 4.00E-05



**:CURSor:H1Position**

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.



**:CURSor:H2Position**

Description	Sets or returns the second horizontal cursor (H2) position.
Syntax	:CURSor:H2Position {<NRf>   ?}
Related Commands	:CURSor:H1Position
Parameter	<NRf> Horizontal Position
Return parameter	Returns the cursor position.
Example	:CURSor:H2Position 1.5E-3 Sets the H2 cursor position to 1.5ms.

**:CURSor:HDELta**

→ Query

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

Set →

**:CURSor:V1Position**

→ Query

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

Set →

**:CURSor:V2Position**

→ Query

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

**:CURSor:VDELta** → Query

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

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

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

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

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

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

:CURSor:XY:RECTangular:Y:POSition<X> Set →  
→ 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.

**:CURSor:HTRACking** Set →  
→ Query

Description	Sets or queries the state of horizontal cursor track.
Syntax	:CURSor:HTRACking {ON OFF} :CURSor:HTRACking?
Example	:CURSor:HTRACking ON :CURSor:HTRACking? ON

**:CURSor:VTRACking** Set →  
→ Query

Description	Sets or queries the state of vertical cursor track.
Syntax	:CURSor:VTRACking {ON OFF} :CURSor:VTRACking?

Example :CURSor:VTRACking ON  
 :CURSor:VTRACking?  
 ON

Set →

→ Query

:CURSor:MARK

Description Sets or queries the state of cursor mark.

Syntax :CURSor:MARK {ON|OFF}  
 :CURSor:MARK?

Example :CURSor:MARK ON  
 :CURSor:MARK?  
 ON

Set →

→ Query

:CURSor:SA:SOURce

Description Sets or queries the SA source of cursor.

Syntax :CURSor:SA:SOURce {SA1|SA2}  
 :CURSor:SA:SOURce?

Related Commands :CURSor:SOURce  
 :CURSor:HUNI  
 :CURSor:HUSE  
 :CURSor:DDT  
 :CURSor:H1Position  
 :CURSor:H2Position  
 :CURSor:HDELta  
 :CURSor:V1Position  
 :CURSor:V2Position

When in SA Mode, the SA source affects the target of the above commands.

Parameter	SA1	Sets cursor SA source as SA1.
	SA2	Sets cursor SA source as SA2.



Example      CURSor:SA:SOURce SA2  
                 CURSor:SA:SOURce?  
                 SA2


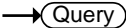
## Display Commands

:DISPlay:INTensity:WAVEform.....	90
:DISPlay:INTensity:GRATICule .....	90
:DISPlay:INTensity:BACKLight .....	91
:DISPlay:INTensity:BACKLight:AUTODim:Enable ...	91
:DISPlay:INTENSITY:BACKLight:AUTODim:TIME ....	91
:DISPlay:PERSistence .....	92
:DISPlay:GRATICule .....	92
:DISPlay:WAVEform .....	93
:DISPlay:OUTPut .....	93
:DISPlay:PNGOutput? .....	93
:DISPlay:WAVEform:COLor.....	94
:DISPlay:RULEr .....	94
:DISPlay:TRANSREADouts .....	94

:DISPlay:INTensity:WAVEform 


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

:DISPlay:INTensity:GRATICule 

  


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 →

:DISPlay:INTensity:BACKLight

→ Query

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 →

:DISPlay:INTensity:BACKLight:AUTODim:ENABLE

→ Query

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.

Set →

:DISPlay:INTENSITY:BACKLight:AUTODim:TIME

→ Query

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 (Set) →  
→ (Query)


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

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

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

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

Return parameter Returns the graticule type.

Example :DISPlay:GRATicule FULL  
Sets the graticule to .

Set →  
 → Query

**: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.	

→ Query

**: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.

→ Query

**:DISPlay:PNGOutput?**

Description	Return the current screen image as PNG image data.
Syntax	:DISPlay:PNGOutput?
Example	:DISPlay:PNGOutput? #516643\89PNG\r\n\1A\r\n\00\00\00\rIHDR...

Set →  
 → Query

**:DISPlay:WAVEform:COLor**

---

Description	Sets or queries the waveform color display format.	
Syntax	:DISPlay:WAVEform:COLor {GRAYscale COLor} :DISPlay:WAVEform:COLor?	
Parameter	GRAY	Sets waveform display as grayscale.
	COLor	Sets waveform display as color.
Example	:DISPlay:WAVEform:COLor GRAYscale :DISPlay:WAVEform:COLor? GRAYSCALE	

Set →  
 → Query

**:DISPlay:RULEr**

---

Description	Sets or queries the state of ruler display.	
Syntax	:DISPlay:RULEr {ON OFF} :DISPlay:RULEr?	
Parameter	ON	Turns on the ruler display.
	OFF	Turns off the ruler display.
Example	:DISPlay:RULEr ON :DISPlay:RULEr? ON	

Set →  
 → Query

**:DISPlay:TRANSREADouts**

---

Description	Sets or queries the state of transparent display.	
Syntax	:DISPlay:TRANSREADouts {ON OFF} :DISPlay:TRANSREADouts?	
Parameter	ON	Turns on the transparent display.
	OFF	Turns off the transparent display.

Example           :DISPlay:TRANSREADouts ON  
                  :DISPlay:TRANSREADouts?  
                  ON

## Hardcopy Commands

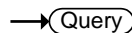
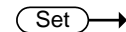
:HARDcopy:START .....	96
:HARDcopy:MODE .....	96
:HARDcopy:PRINTINKSaver .....	97
:HARDcopy:SAVEINKSaver .....	97
:HARDcopy:SAVEFORMat .....	97
:HARDcopy:ASSIGN .....	98

### :HARDcopy:START



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

### :HARDcopy:MODE

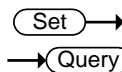


Description	Sets or queries whether hardcopy is set to print or save.
Syntax	:HARDcopy:MODE { PRINT   SAVE   ? }
Related Commands	:HARDcopy:START
Parameter	PRINT      Print mode SAVE        Save mode
Return parameter	Returns the mode.(PRINT/SAVE)



Example :HARDcopy:MODE PRINT  
Sets hardcopy to print.

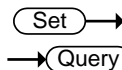
**:HARDcopy:PRINTINKSaver**



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.

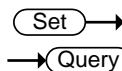
**:HARDcopy:SAVEINKSaver**



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	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**

Set →

→ Query

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

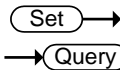
## Measure Commands

---

:MEASure:GATing.....	100
:MEASure:SOURce<X>.....	101
:MEASure:METHod .....	101
:MEASUrement:REFLevel:PERCent:HIGH.....	102
:MEASUrement:REFLevel:PERCent:LOW .....	102
:MEASUrement:REFLevel:PERCent:MID .....	102
:MEASUrement:REFLevel:PERCent:MID2 .....	103
:MEASUrement:REFLevel:ABSolute:HIGH.....	103
:MEASUrement:REFLevel:ABSolute:LOW .....	103
:MEASUrement:REFLevel:ABSolute:MID .....	104
:MEASUrement:REFLevel:ABSolute:MID2 .....	104
:MEASUrement:REFLevel:METHod .....	104
:MEASure:FALL.....	105
:MEASure:FOVShoot .....	105
:MEASure:FPReshoot .....	106
:MEASure:FREQuency .....	107
:MEASure:NWIDth .....	107
:MEASure:PDUTy.....	108
:MEASure:PERiod .....	108
:MEASure:PWIDth .....	109
:MEASure:RISe.....	109
:MEASure:ROVShoot.....	110
:MEASure:RPReshoot .....	110
:MEASure:PPULSE.....	111
:MEASure:NPULSE.....	112
:MEASure:PEDGE .....	112
:MEASure:NEDGE .....	113
:MEASure:AMPlitude.....	113
:MEASure:MEAN .....	114
:MEASure:CMEan .....	114
:MEASure:HIGH.....	115
:MEASure:LOW.....	115

:MEASure:MAX ..... 116  
 :MEASure:MIN ..... 117  
 :MEASure:PK2PK ..... 117  
 :MEASure:RMS ..... 118  
 :MEASure:CRMS ..... 118  
 :MEASure:AREa ..... 119  
 :MEASure:CARea ..... 119  
 :MEASure:FRRDelay ..... 120  
 :MEASure:FRFDelay ..... 121  
 :MEASure:FFRDelay ..... 121  
 :MEASure:FFFDelay ..... 122  
 :MEASure:LRRDelay ..... 122  
 :MEASure:LFRFDelay ..... 123  
 :MEASure:LFRFDelay ..... 124  
 :MEASure:LFFFDelay ..... 124  
 :MEASure:PHase ..... 125  
 :MEASure:PFLicker ..... 125  
 :MEASure:FLicker ..... 126

**: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<X>** 


Description	Sets or queries the measurement source for source1 or source2.	
Syntax	:MEASure:SOURce<X> { CH1   CH2   CH3   CH4   MATH   ? }	
Parameter	<X>	Source1 or source2
	CH1~CH4	Channel 1 to 4
	MATH	Math
Return parameter	Returns the source (CH1, CH2, CH3, CH4, MATH)	
Example	:MEASure:SOURce1 CH1 Sets source1 to channel 1.	

**:MEASure:METHOD** 


Description	Sets or queries the method used to determine the High-Low measurement values.	
Syntax	:MEASure:METHOD { AUTo   HISTogram   MINMax   ? }	
Parameter	AUTo	Set to auto.
	HISTogram	Set to the Histogram method.
	MINMax	Set to the Min-Max method.
Return parameter	Returns the measurement method (AUTO, HISTOGRAM, MINMAX)	
Example	:MEASure:METHOD: AUTo Set the measurement method to auto.	

**:MEASUrement:REFLevel:PERCent:HIGH** 
 Set →  
 Query

Description	Sets or queries the high reference level as a percentage.	
Syntax	:MEASUrement:REFLevel:PERCent:HIGH {<NRF>   ?}	
Parameter	<NRF>	0 - 100%
Return parameter	Returns the high reference level	
Example	:MEASUrement:REFLevel:PERCent:HIGH 50.1 Set the high reference level to 50.1%.	

**:MEASUrement:REFLevel:PERCent:LOW** 
 Set →  
 Query

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

**:MEASUrement:REFLevel:PERCent:MID** 
 Set →  
 Query

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

:MEASUrement:REFLevel:PERCent:MiD2  

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%.

:MEASUrement:REFLevel:ABSolute:HiGH  

Description	Sets or returns the value for the high reference level.
Syntax	:MEASUrement:REFLevel:ABSolute:HiGH <NRf> :MEASUrement:REFLevel:ABSolute:HiGH?
Parameter	<NRf> This is the high reference level, ranging from -100M(-1e+8) to 100M(1e+8).
Example	MEASUrement:REFLevel:ABSolute:HiGH 0.06 MEASUrement:REFLevel:ABSolute:HiGH? 6.00e-02

:MEASUrement:REFLevel:ABSolute:LOW  

Description	Sets or returns the value for the low reference level.
Syntax	:MEASUrement:REFLevel:ABSolute:LOW <NRf> :MEASUrement:REFLevel:ABSolute:LOW?
Parameter	<NRf> This is the low reference level, ranging from -100M(-1e+8) to 100M(1e+8).

Example            MEASUrement:REFLevel:ABSolute:LOW 100  
                       MEASUrement:REFLevel:ABSolute:LOW?  
                       1.00e+02

Set →

:MEASUrement:REFLevel:ABSolute:MID

→ Query

Description       Sets or returns the value for the middle reference level.

Syntax             :MEASUrement:REFLevel:ABSolute:MID <NRf>  
                       :MEASUrement:REFLevel:ABSolute:MID?

Parameter        <NRf>    This is the middle reference level, ranging from -100M(-1e+8) to 100M(1e+8)

Example            MEASUrement:REFLevel:ABSolute:MID 58  
                       MEASUrement:REFLevel:ABSolute:MID?  
                       5.80e+01

Set →

:MEASUrement:REFLevel:ABSolute:MID2

→ Query

Description       Sets or returns the value for the middle reference level.

Syntax             :MEASUrement:REFLevel:ABSolute:MID2 <NRf>  
                       :MEASUrement:REFLevel:ABSolute:MID2?

Parameter        <NRf>    This is the middle reference level, ranging from -100M(-1e+8) to 100M(1e+8).

Example            MEASUrement:REFLevel:ABSolute:MID2 -40  
                       MEASUrement:REFLevel:ABSolute:MID2?  
                       -4.00e+01

Set →

:MEASUrement:REFLevel:METHod

→ Query


Description       Sets or returns the reference level units.



Syntax	:MEASUrement:REFLevel:METHOD {ABSolute PERCent} :MEASUrement:REFLevel:METHOD?	
Parameter	ABSolute	Set the reference level units as absolute.
	PERCent	Set the reference level units as percent.
Example	MEASUrement:REFLevel:METHOD ABSolute :MEASUrement:REFLevel:METHOD? ABSOLUTE	

**: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>	Returns the fall overshoot as a percentage
	Chan Off	Indicates the source channel is not activated.

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

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

**:MEASure:FPReshoot** → Query


Description	Returns fall preshoot amplitude.	
Syntax	:MEASure:FPReshoot{?}	
Related Commands	:MEASure:SOURce<X>	
Returns	Returns the fall preshoot as <NR3>.	
Return parameter	<NR3>	Returns the fall preshoot as a percentage.
	Chan Off	Indicates the source channel is not activated.

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

**Example** :MEASure:SOURce1 CH1  
 :MEASure:FPReshoot?  
 Selects Channel 1, and then measures the fall preshoot.

**:MEASure:FREQuency** → Query

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

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


**Example**

```
:MEASure:SOURce1 CH1
:MEASure:FREQuency?
>1.0E+3
```

Selects Channel 1, and then measures the frequency.

**:MEASure:NWIDth** → Query

Description	Returns the first negative pulse width timing.	
Syntax	:MEASure:NWIDth{?}	
Related Commands	:MEASure:SOURce<X>	
Return parameter	<NR3>	Returns the negative pulse width in seconds.
	Chan Off	Indicates the source channel is not activated.

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

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

**:MEASure:PDUTy** → Query

Description Returns the positive duty cycle ratio as percentage.

Syntax :MEASure:PDUTy{?}

Related commands :MEASure:SOURce<X>

Return parameter	<b>&lt;NR3&gt;</b>	Returns the positive duty ratio.
	Chan Off	Indicates the source channel is not activated.



Note

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

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

**:MEASure:PERiod** → Query

Description Returns the period.

Syntax :MEASure:PERiod{?}

Related Commands :MEASure:SOURce<X>

Return parameter	<b>&lt;NR3&gt;</b>	Returns the period.
------------------	--------------------	---------------------

Chan Off Indicates the source channel is not activated.



Note

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

Example

```
:MEASure:SOURce1 CH1
```

```
:MEASure:PERiod?
```

```
1.0E-3
```

Selects Channel 1, and then measures the period.

### :MEASure:PWIDth



Description

Returns the first positive pulse width.

Syntax

```
:MEASure:PWIDth{?}
```

Related

```
:MEASure:SOURce<X>
```

Commands

Return parameter

```
<NR3>
```

Returns the positive pulse width.

Chan Off

Indicates the source channel is not activated.



Note

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

Example

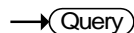
```
:MEASure:SOURce1 CH1
```

```
:MEASure:PWIDth?
```

```
5.0E-6
```

Selects Channel 1, and then measures the positive pulse width.

### :MEASure:RISe



Description

Returns the first pulse rise time.

Syntax

```
:MEASure:RISe{?}
```

Related

```
:MEASure:SOURce<X>
```

Commands

Return parameter	<NR3>	Returns the rise time.
	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:RISe? 8.5E-6 Selects Channel 1, and then measures the rise time.
---------	---

**:MEASure:ROVShoot**



Description	Returns the rising overshoot over the entire waveform in percentage.	
Syntax	:MEASure:ROVShoot{?}	
Related Commands	:MEASure:SOURce<X>	
Return parameter	<NR3>	Returns the overshoot.
	Chan Off	Indicates the source channel is not activated.



Note


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

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

**:MEASure:RPReshoot**




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	<pre>:MEASure:SOURce1 CH1 :MEASure:PPULSE? 6.000E+00</pre> Selects Channel 1, and then measures the number of positive pulses.	

**:MEASure:NPULSE**

→ **Query**

Description	Returns the number of negative pulses.	
Syntax	:MEASure:NPULSE{?}	
Related Commands	:MEASure:SOURce<X>	
Return parameter	<NR3>	Returns the number of negative pulses.
	Chan Off	Indicates the source channel is not activated.



Note

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

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

**:MEASure:PEDGE**

→ **Query**

Description	Returns the number of positive edges.	
Syntax	:MEASure:PEDGE{?}	
Related Commands	:MEASure:SOURce<X>	
Return parameter	<NR3>	Returns the number of positive edges.
	Chan Off	Indicates the source channel is not activated.



Note

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

Example	:MEASure:SOURce1 CH1 :MEASure:PEDGE?	
---------	---	--



1.100E+01

Selects Channel 1, and then measures the number of positive edges.

**:MEASure:NEDGE**

→ Query

Description	Returns the number of negative edges.	
Syntax	:MEASure:NEDGE{?}	
Related Commands	:MEASure:SOURce<X>	
Return parameter	<NR3>	Returns the number of negative edges.
	Chan Off	Indicates the source channel is not activated.



Note

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

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

**:MEASure:AMPlitude**

→ Query

Description	Returns the amplitude difference between the Vhigh-Vlow.	
Syntax	:MEASure:AMPlitude{?}	
Related Commands	:MEASure:SOURce<X>	
Return parameter	<NR3>	Returns the amplitude.
	Chan Off	Indicates the source channel is not activated.



Note

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

Example

```
:MEASure:SOURce1 CH1
```

```
:MEASure:AMPLitude?
```

```
3.76E-3
```

Selects Channel 1, and then measures the amplitude.

### :MEASure:MEAN



Description

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

Syntax

```
:MEASure:MEAN{?}
```

Related

```
:MEASure:SOURce<X>
```

Commands

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



Description

Returns the mean voltage/current of one full period.

Syntax

```
:MEASure:CMEan{?}
```

Related

```
:MEASure:SOURce<X>
```

Commands

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

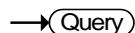


Note

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



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



Description	Returns the global low voltage/current.
Syntax	:MEASure:LOW{?}

Related Commands :MEASure:SOURce<X>

Return parameter	<b>&lt;NR3&gt;</b>	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** → 

**Description** Returns the maximum amplitude.

**Syntax** :MEASure:MAX{?}

Related Commands :MEASure:SOURce<X>

Return parameter	<b>&lt;NR3&gt;</b>	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**



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

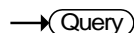
:MEASure:SOURce1 CH1

:MEASure:AREa?

1.958E-03

Selects Channel 1, and then measures the area.

**:MEASure:CARea**



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** → 

**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>

<b>Return parameter</b>	<b>&lt;NR3&gt;</b>	Returns the delay.
	<b>Chan Off</b>	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:FFRDelay{?}

**Related Commands** :MEASure:SOURce<X>

<b>Return parameter</b>	<b>&lt;NR3&gt;</b>	Returns the delay.
	<b>Chan Off</b>	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.

## :MEASure:PFLIcker


→ Query

Description       Returns the % flicker of times.

Syntax            :MEASure:PFLI?

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:PFLI ?  
5.950E+01  
Selects Channel 1 as the source, and then measures the % flicker of times.

**:MEASure:FLIcker** → 

<b>Description</b>	Returns the flicker idx of times.
<b>Syntax</b>	:MEASure:FLI?
<b>Related Commands</b>	: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:FLI ?  
2.870E-01  
Selects Channel 1 as the source, and then measures the flicker idx of times.

## Measurement Commands

:MEASUrement:MEAS<X>:SOURCE<X> .....	127
:MEASUrement:MEAS<X>:TYPE .....	128
:MEASUrement:MEAS<X>:STATE .....	129
:MEASUrement:MEAS<X>:VALue .....	129
:MEASUrement:MEAS<X>:MAXimum .....	130
:MEASUrement:MEAS<X>:MEAN .....	131
:MEASUrement:MEAS<X>:MINimum .....	131
:MEASUrement:MEAS<X>:STDdev .....	132
:MEASUrement:STATIstics:MODE .....	133
:MEASUrement:STATIstics:WEIghting .....	133
:MEASUrement:STATIstics .....	133
:MEASUrement:INDICators:STAT .....	134
:MEASUrement:INDICators:HORIZ<x>? .....	134
:MEASUrement:INDICators:VERT<x>? .....	135
:MEASUrement:INDICators:NUMHORZ? .....	135
:MEASUrement:INDICators:NUMVERT? .....	135


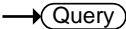
:MEASUrement:MEAS<X>:SOURCE<X> 


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   D0   D1   D2   D3   D4   D5   D6   D7   D8   D9   D10   D11   D12   D13   D14   D15   ? }	
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
	D0~D15	Digital source D0~D15
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 



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 | PFLIcker | FLIcker | 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** 



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** 

Description	Returns the measurement results for the selected measurement. This is a statistics related command.	
Syntax	:MEASUrement:MEAS<X>:VALue?	
Related Commands	:MEASure:SOURce<X>	
Parameter	MEAS<X>	The automatic measurement number from 1 to 8.
Return parameter	<NR3>	Returns the measurement for the selected measurement number.



Note

The measurement source(s), measurement number, measurement type and measurement state must first be set before a measurement result can be returned.

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

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

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

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

Syntax :MEASUrement:MEAS<X>:MAXimum?

Related Commands :MEASUrement:STATIstics:MODE

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

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

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

Returns the maximum measurement result for measurement number 3.

**:MEASUrement:MEAS<X>:MEAN** → Query

Description	Returns the mean measurement results for the selected measurement from the last time the statistics were reset. This is a statistics related command.	
Syntax	:MEASUrement:MEAS<X>:MEAN?	
Related Commands	:MEASUrement:STATIstics:MODE	
Parameter	MEAS<X>	The automatic measurement number from 1 to 8.
Return parameter	<NR3>	Returns the measurement for the selected measurement number.

**Example**

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

Returns the mean measurement result for measurement number 3.

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

Description	Returns the minimum measurement results for the selected measurement from the last time the statistics were reset. This is a statistics related command.	
Syntax	:MEASUrement:MEAS<X>:MINIum?	
Related Commands	:MEASUrement:STATIstics:MODE	

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

Example

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

Returns the minimum measurement result for measurement number 3.

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

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

Syntax :MEASUrement:MEAS<X>:STDdev?

Related Commands :MEASUrement:STATIstics:MODE

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

Example

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

Returns the standard deviation for measurement number 3.

:MEASUREMENT:STATISTICS:MODE  

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  

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 

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

**:MEASUrement:INDICators:STAT** 



Description	Set or query the measurement indicator's state.	
Syntax	:MEASUrement:INDICators:STAT {OFF MEAS<x>} :MEASUrement:INDICators:STAT?	
Parameter	OFF	Turn off the indicator.
	MEAS<x>	Set the indicator source. <x>:1~8.
Example	:MEASUrement:INDICators:STAT OFF :MEASUrement:INDICators:STAT? OFF :MEASUrement:INDICators:STAT MEAS2 :MEASUrement:INDICators:STAT? MEAS2	

**:MEASUrement:INDICators:HORIZ<x>?** 

Description	Query the position of measurement indicator's horizontal track.	
Syntax	:MEASUrement:INDICators:HORIZ<x>? {PRECise}	
Parameter	HORIZ<x>	The horizontal track. <x>:1 or 2.
	PRECise	Display more digit for the return value.
Example	:MEASUrement:INDICators:HORIZ1? 3.120e+00 :MEASUrement:INDICators:HORIZ1? PRECise 3.120000e+00	

**:MEASUrement:INDICators:VERT<x>?** → **Query**

Description	Query the position of measurement indicator's vertical track.
Syntax	:MEASUrement:INDICators:VERT<x>? {PRECise}
Parameter	<p>VERT&lt;x&gt; The vertical track. &lt;x&gt;:1 or 2.</p> <p>PRECise Display more digit for the return value.</p>
Example	<pre>:MEASUrement:INDICators:VERT1? -2.135e-02 :MEASUrement:INDICators:VERT1? PRECise -2.135000e-02</pre>

**:MEASUrement:INDICators:NUMHORZ?** → **Query**

Description	Query the number of measurement indicator's horizontal tracks currently being displayed.
Syntax	:MEASUrement:INDICators:NUMHORZ?
Example	<pre>:MEASUrement:INDICators:NUMHORZ? 2</pre>

**:MEASUrement:INDICators:NUMVERT?** → **Query**

Description	Query the number of measurement indicator's vertical tracks currently being displayed.
Syntax	:MEASUrement:INDICators:NUMVERT?
Example	<pre>:MEASUrement:INDICators:NUMVERT? 1</pre>

## Reference Commands

:REF<X>:DISPlay .....	136
:REF<X>:TIMebase:POSition .....	136
:REF<X>:TIMebase:SCALe .....	137
:REF<X>:OFFSet .....	137
:REF<x>:SCALe .....	138

:REF<X>:DISPlay 
 Set →  
 →  Query

**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| ? }

<b>Parameter</b>	<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 
 Set →  
 →  Query

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


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

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

<b>Parameter</b>	<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.	

: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.	

: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	<NR3>	Returns the reference waveform vertical position.

Example :REF1:OFFSet -5.000E-2

Selects reference 1, and then sets the vertical position to -50mV/mA.

:REF<x>:SCALE

Set →

→ Query

Description Sets or returns the selected reference waveform vertical scale.

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

Related commands :REF<X>:DISPlay

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

Return parameter	<NR3>	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).
-------------	---

---

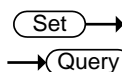
Syntax	:FORCe
--------	--------

---

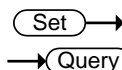
## Timebase Commands

:TIMebase:EXPand .....	141
:TIMebase:POSition .....	141
:TIMebase:SCALe.....	142
:TIMebase:MODE .....	142
:TIMebase:WINDow:POSition .....	142
:TIMebase:WINDow:SCALe .....	143
:PLAYStop .....	143

### :TIMebase:EXPand



Description	Sets or queries the horizontal expansion mode.	
Syntax	:TIMebase:EXPand {CENTer TRIGger ?}	
Parameter/Return parameter	CENTer	Expand from the center of the display.
	TRIGger	Expand from the trigger point.
Example	:TIMebase:EXPand TRIGger Sets the expansion point to the trigger point.	



### :TIMebase:POSition

Description	Sets or queries the horizontal position.	
Syntax	:TIMebase:POSition {<NRf>   ?}	
Parameter	<NRf>	Horizontal position
Return parameter	<NR3>	Returns the horizontal position
Example	:TIMebase:POSition 5.00E-4 Sets the horizontal position as 500us.	

Set →  
 → Query

**:TIMEbase:SCALE**

---

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

Set →  
 → Query

**:TIMEbase:MODE**

---

Description	Sets or queries the time base mode. The time base mode determines the display view window on the scope.						
Syntax	:TIMEbase:MODE {MAIN   WINDow   XY   ?}						
Parameter	<table border="0" style="width: 100%;"> <tr> <td style="width: 15%; padding-right: 10px;">MAIN</td> <td style="padding-right: 10px;">Sets the time base mode to the main screen.</td> </tr> <tr> <td style="padding-right: 10px;">WINDow</td> <td style="padding-right: 10px;">Sets the time base mode to the zoom window.</td> </tr> <tr> <td style="padding-right: 10px;">XY</td> <td style="padding-right: 10px;">Sets the time base mode to the XY display.</td> </tr> </table>	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.
MAIN	Sets the time base mode to the main screen.						
WINDow	Sets the time base mode to the zoom window.						
XY	Sets the time base mode to the XY display.						
Return parameter	Returns the time base mode (MAIN, WINDOW, XY)						
Example	:TIMEbase:MODE MAIN Sets the time base mode to the main mode.						

Set →  
 → Query

**:TIMEbase:WINDow:POSition**

---

Description	Sets or queries the zoom horizontal position.
Syntax	:TIMEbase:WINDow:POSition {<NRf>   ?}
Related commands	:TIMEbase:MODE

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

Set →  
 → Query

**: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.	

**:PLAYStop** Set →

Description	Set the zoom play/stop or play the current segment in segments mode.	
Syntax	:PLAYStop {ON OFF} :PLAYStop?	
Parameter	ON	Play
	OFF	Stop
Example	:PLAYStop ON :PLAYStop? ON	

## Trigger Commands

---

:TRIGger:FREQuency.....	146
:TRIGger:TYPe .....	146
:TRIGger:SOURce .....	147
:TRIGger:COUPle.....	147
:TRIGger:NREJ .....	148
:TRIGger:MODe.....	148
:TRIGger:HOLDoff.....	148
:TRIGger:LEVel.....	149
:TRIGger:HLEVel .....	149
:TRIGger:LLEVel.....	150
:TRIGger:EDGe:SLOP.....	150
:TRIGger:DElay:SLOP .....	151
:TRIGger:DElay:TYPe.....	151
:TRIGger:DElay:TIMe.....	151
:TRIGger:DElay:EVENT.....	152
:TRIGger:DElay:LEVel .....	152
:TRIGger:PULSEWidth:POLarity .....	152
:TRIGger:RUNT:POLarity .....	153
:TRIGger:RUNT:WHEn.....	153
:TRIGger:RUNT:TIMe.....	154
:TRIGger:RISEFall:SLOP.....	154
:TRIGger:RISEFall:WHEn .....	155
:TRIGger:RISEFall:TIMe .....	155
:TRIGger:VIDeo:TYPe .....	156
:TRIGger:VIDeo:FIELD .....	156
:TRIGger:VIDeo:LINE .....	157
:TRIGger:VIDeo:POLarity .....	157
:TRIGger:PULSe:WHEn .....	157
:TRIGger:PULSe:TIMe .....	158
:TRIGger:TIMEOut:WHEn.....	158
:TRIGger:TIMEOut:TIMER .....	159
:TRIGger:ALTErnate .....	159



:TRIGger:STATe .....	160
:TRIGger:EXTERnal:PROBe:TYPe .....	160
:TRIGger:EXTERnal:PROBe:RATio .....	161
:TRIGger:BUS:TYPe .....	161
:TRIGger:BUS:THReshold:CH<x> .....	162
:TRIGger:BUS:B1:I2C:CONDition .....	162
:TRIGger:BUS:B1:I2C:ADDRes:MODE .....	163
:TRIGger:BUS:B1:I2C:ADDRes:TYPe .....	163
:TRIGger:BUS:B1:I2C:ADDRes:VALue .....	164
:TRIGger:BUS:B1:I2C:ADDRes:DIRection .....	165
:TRIGger:BUS:B1:I2C:DATA:SIZE .....	165
:TRIGger:BUS:B1:I2C:DATA:VALue .....	166
:TRIGger:BUS:B1:UART:CONDition .....	166
:TRIGger:BUS:B1:UART:RX:DATA:SIZE .....	167
:TRIGger:BUS:B1:UART:RX:DATA:VALue .....	168
:TRIGger:BUS:B1:UART:TX:DATA:SIZE .....	168
:TRIGger:BUS:B1:UART:TX:DATA:VALue .....	169
:TRIGger:BUS:B1:SPI:CONDition .....	170
:TRIGger:BUS:B1:SPI:DATA:SIZE .....	170
:TRIGger:BUS:B1:SPI:DATA:MISO:VALue .....	171
:TRIGger:BUS:B1:SPI:DATA:MOSI:VALue .....	171
:TRIGger:BUS:B1:CAN:CONDition .....	172
:TRIGger:BUS:B1:CAN:FRAMeType .....	173
:TRIGger:BUS:B1:CAN:IDentifier:MODE .....	173
:TRIGger:BUS:B1:CAN:IDentifier:VALue .....	174
:TRIGger:BUS:B1:CAN:IDentifier:DIRection .....	174
:TRIGger:BUS:B1:CAN:DATA:QUALifier .....	175
:TRIGger:BUS:B1:CAN:DATA:SIZE .....	176
:TRIGger:BUS:B1:CAN:DATA:VALue .....	176
:TRIGger:BUS:B1:LIN:CONDition .....	177
:TRIGger:BUS:B1:LIN:DATA:QUALifier .....	178
:TRIGger:BUS:B1:LIN:DATA:SIZE .....	178
:TRIGger:BUS:B1:LIN:DATA:VALue .....	179
:TRIGger:BUS:B1:LIN:ERRTYPe .....	180
:TRIGger:BUS:B1:LIN:IDentifier:VALue .....	180

:TRIGger:BUS:B1:PARallel:VALue ..... 181  
 :TRIGger:LOGic:INPut:CLOCK:SOURce..... 181  
 :TRIGger:LOGic:INPut:CLOCK:EDGE ..... 182  
 :TRIGger:LOGic:FUNCTion ..... 182  
 :TRIGger:LOGic:PATtern ..... 183  
 :TRIGger:LOGic:PATtern:INPut:D<x> ..... 183  
 :TRIGger:LOGic:PATtern:DELtetime..... 184  
 :TRIGger:LOGic:PATtern:WHEn ..... 184

**:TRIGger:FREQuency**

→ Query

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

Set →

**:TRIGger:TYPE**

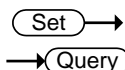
→ Query

Description	Sets or queries the trigger type.	
Syntax	:TRIGger:TYPE {EDGE   DELay   PULSEWidth   VIDeo   RUNT   RISEFall   LOGic   BUS   TIMEOut   ? }	
Parameter	EDGE	Edge trigger
	DELay	Delay trigger
	PULSEWidth	Pulse width trigger
	VIDeo	Video trigger
	RUNT	Runt trigger
	RISEFall	Rise and fall trigger
	LOGic	Logic trigger

BUS	Bus trigger
TIMEOut	Timeout trigger

Return parameter Returns the trigger type.

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



### :TRIGger:SOURce

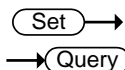
Description Sets or queries the trigger source.

Syntax :TRIGger:SOURce  
{ CH1 | CH2 | CH3 | CH4 | EXT | LINE | D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | D8 | D9 | D10 | D11 | D12 | D13 | D14 | D15 | ? }

Parameter	CH1 to CH4	Channel 1 to channel 4
	EXT	External source
	LINE	AC Line
	D0~D15	Digital channels D0~D15

Return parameter Returns the trigger source.

Example :TRIGger:SOURce CH1  
Sets the trigger source to channel 1.



### :TRIGger:COUple

Description Sets or queries the trigger coupling.

Note Applicable for edge and delay triggers only.

Syntax :TRIGger:COUple {AC | DC | HF | LF | ?}

Parameter	AC	AC mode
	DC	DC mode
	HF	High frequency rejection
	LF	Low frequency rejection

Return parameter Returns the trigger coupling.

Example :TRIGger:COUPlE AC  
Sets the trigger coupling to AC.

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:NREJ ON  
Turns noise rejection on.

Set →

→ Query

**:TRIGger:MODE**

Description Sets or queries the trigger mode.

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

Parameter	AUTo	Auto trigger (Untriggered roll)
	NORMAl	Normal trigger

Return parameter Returns the trigger mode.

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

Set →

→ Query

**:TRIGger:HOLDoff**

Description Sets or queries the holdoff time.

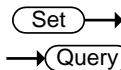
Syntax :TRIGger:HOLDoff {<NRf> | ?}

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

Return parameter <NR3> Returns the trigger holdoff time.

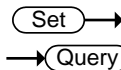
Example :TRIGger:HOLDoff 1.00E-8  
Sets the trigger holdoff time to 10ns.

**:TRIGger:LEVel**



Description	Sets or queries the level.	
Note	Not applicable to Pulse Runt and Rise & Fall triggers.	
Syntax	:TRIGger:LEVel {TTL   ECL   SETTO50   <NRf>   ?}	
Related commands	:TRIGger:TYPe	
Parameter	<NRf>	Trigger level value.
	TTL	Sets the trigger level to TTL.
	ECL	Sets the trigger level to ECL.
	SETTO50	Sets the trigger level to the User level (50% by default).
Return parameter	<NR3>	Returns the trigger level.
Example1	:TRIGger:LEVel TTL Sets the trigger to TTL.	
Example2	:TRIGger:LEVel 3.30E-1 Sets the trigger level to 330mV/mA.	

**:TRIGger:HLEVel**



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

Example :TRIGger:HLEVel 3.30E-1  
 Sets the trigger high level to 330mV/mA.

Set →

:TRIGger:LLEVel

→ Query

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 →

:TRIGger:EDGE:SLOP

→ Query

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	<table border="0" style="width: 100%;"> <tr> <td style="width: 150px;">RISe</td> <td>Rising slope</td> </tr> <tr> <td>FALL</td> <td>Falling slope</td> </tr> <tr> <td>EITHer</td> <td>Either rising or falling slope</td> </tr> </table>	RISe	Rising slope	FALL	Falling slope	EITHer	Either rising or falling slope
RISe	Rising slope						
FALL	Falling slope						
EITHer	Either rising or falling slope						
Return parameter	Returns the trigger slope.						
Example	:TRIGger:DElay:SLOP FALL Sets the trigger slope to falling.						

Set →  
→ Query

---

**:TRIGger:DElay:TYPe**

---

Description	Sets or queries the trigger delay type.				
Syntax	:TRIGger:DElay:TYPe {TImE   EVEnt   ? }				
Related commands	:TRIGger:TYPe				
Parameter	<table border="0" style="width: 100%;"> <tr> <td style="width: 150px;">TImE</td> <td>Sets the delay type to time.</td> </tr> <tr> <td>EVEnt</td> <td>Sets the delay type to event.</td> </tr> </table>	TImE	Sets the delay type to time.	EVEnt	Sets the delay type to event.
TImE	Sets the delay type to time.				
EVEnt	Sets the delay type to event.				
Return parameter	Returns the trigger delay type.				
Example	:TRIGger:DElay:TYPe TImE Sets the delay type to time delay.				

Set →  
→ Query

---

**:TRIGger:DElay:TImE**

---

Description	Sets or queries the delay time value.
Syntax	:TRIGger:DElay:TImE {<NRf>   ? }

Related commands :TRIGger:DElay:TYPe

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

Return parameter <NR3> Returns the delay time.

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

Set →

→ Query

**:TRIGger:DElay:EVENT**

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

Syntax :TRIGger:DElay:EVENT {<NR1> | ?}

Related commands :TRIGger:DElay:TYPe

Parameter <NR1> 1~65535 events

Return parameter <NR1> Returns the number of events.

Example :TRIGger:DElay:EVENT 2  
Sets the number of events to 2.

Set →

→ Query

**:TRIGger:DElay:LEVel**

Description Sets or queries the trigger delay level.

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

Parameter <NRf> Delay trigger level

Return parameter <NR3> Returns the delay trigger.

Example :TRIGger:DElay:LEVel 5.00E-3  
Sets the delay trigger level to 5mV/mA.

Set →

→ Query

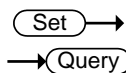
**:TRIGger:PULSEWidth:POLarity**

Description Sets or queries the pulse width trigger polarity.

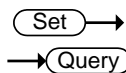


Syntax	:TRIGger:PULSEWidth:POLarity {POSitive   NEGative   ?}	
Related commands	:TRIGger:TYPe	
Parameter	POSitive	Positive polarity
	NEGative	Negative polarity
Return parameter	Returns the pulse width polarity.	
Example	:TRIGger:PULSEWidth:POLarity POSitive Sets the pulse width polarity to positive.	

**:TRIGger:RUNT:POLarity**



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



**:TRIGger:RUNT:WHEn**

Description	Sets or queries the Pulse Runt trigger conditions.	
Syntax	:TRIGger:RUNT:WHEn {MOREthan  LESSthan   Equal   UNEQual   ? }	
Related commands	:TRIGger:TYPe :TRIGger:RUNT:TIME	

Parameter	MOREthan	>
	LESSthan	<
	Equal	=
	UNEQual	≠

Return parameter Returns the pulse runt trigger condition.

Example :TRIGger:RUNT:WHEn UNEQual  
Sets the Pulse Runt trigger condition to unequal (≠).

Set →

→ Query

### :TRIGger:RUNT:TIME

Description Sets or queries the Pulse Runt trigger time.

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

Related commands :TRIGger:TYPe  
:TRIGger:RUNT:WHEn

Parameter	<NRf>	Pulse runt time (4nS to 10S)
Return Parameter	<NR3>	Returns the runt time in seconds.

Example :TRIGger:RUNT:TIME 4.00E-5  
Sets the runt time to 40.0uS.

Set →

→ Query

### :TRIGger:RISEFall:SLOP

Description Sets or queries the Rise & Fall slope.

Syntax :TRIGger:RISEFall:SLOP {RISe | FALL | EITHer | ? }

Parameter	RISe	Rising slope
	FALL	Falling slope
	EITHer	Either rising or falling slope

Return parameter Returns the rise & fall slope.

Example :TRIGger:RISEFall:SLOP RISe  
Sets the Rise & Fall slope to rising.

**:TRIGger:RISEFall:WHEN** 


Description	Sets or queries the rise/fall trigger conditions.								
Syntax	:TRIGger:RISEFall:WHEN {MOREthan   LESSthan   EQual   UNEQual   ? }								
Related commands	:TRIGger:TYPE :TRIGger:RISEFall:TIME								
Parameter	<table border="0"> <tr><td>MOREthan</td><td>&gt;</td></tr> <tr><td>LESSthan</td><td>&lt;</td></tr> <tr><td>Equal</td><td>=</td></tr> <tr><td>UNEQual</td><td>≠</td></tr> </table>	MOREthan	>	LESSthan	<	Equal	=	UNEQual	≠
MOREthan	>								
LESSthan	<								
Equal	=								
UNEQual	≠								
Return parameter	Returns the rise/fall trigger condition.								
Example	:TRIGger:RISEFall:WHEN UNEQual Sets the Rise and Fall trigger condition to unequal (≠).								

**:TRIGger:RISEFall:TIME** 


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.

Set →

→ Query

**:TRIGger:VIDeo:TYPe**

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**

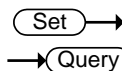
Description	Sets or queries the video trigger field.	
Syntax	:TRIGger:VIDeo:FIELD { FIELD1   FIELD2   ALLFields   ALLLines   ? }	
Related commands	:TRIGger:TYPe	
Parameter	FIELD1	Trigger on field 1
	FIELD2	Trigger on field 2
	ALLFields	Trigger on all fields

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.

:TRIGger:VIDeo:LINE



Description Sets or queries the video trigger line.

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

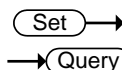
Related commands :TRIGger:TYPE

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 | ? }

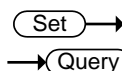
Related commands :TRIGger:TYPE

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 (≠).

Set →

→ Query

**:TRIGger:PULSe:TIME**

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

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

Set →

→ Query

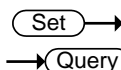
**:TRIGger:TIMEOut:WHEn**

Description	Sets or queries the timeout trigger condition.	
Syntax	:TRIGger:TIMEOut:WHEn {HIGH LOW EITHER ?}	
Related commands	:TRIGger:TIMEOut:TIMER	

Parameter	HIGH	Signal is high.
	LOW	Signal is low.
	EITHer	Signal is high or low.

Return parameter Returns the timeout condition (HIGH, LOW, EITHER).

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



**:TRIGger:TIMEOut:TIMER**

Description Sets or returns timeout trigger time.

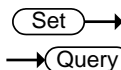
Syntax :TRIGger:TIMEOut:TIMER {<NRf> | ? }

Related commands :TRIGger:TIMEOut:WHEn

Parameter	<NRf>	Timeout time. (4nS to 10S).
-----------	-------	-----------------------------

Return parameter Returns the timeout time as <NR3>.

Example :TRIGger:TIMEOut:TIMER?  
8.960e-05



**:TRIGger:ALTErnate**

Description Sets alternating between source triggers on or off or queries its state.

Syntax :TRIGger:ALTErnate {OFF | ON |?}

Parameter	OFF	Alternate off
	ON	Alternate on

Return parameter Returns the Alternate trigger status (ON, OFF).

Example :TRIGger:ALTErnate ON  
Turns on alternating between source triggers.

**:TRIGger:STATe**

→ Query

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 →

**:TRIGger:EXTERnal:PROBE:TYPE**

→ Query

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

Return parameter Returns the probe type.



Example :TRIGger:EXTERnal:PROBe:TYPe?  
CURRENT

**:TRIGger:EXTERnal:PROBE:RATio**

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	12C	I <sup>2</sup> C mode
	SPI	SPI mode
	UART	UART mode
	CAN	CAN mode
	PARALLEL	PARALLEL 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 <sup>2</sup> C trigger conditions.	
Syntax	:TRIGger:BUS:B1:I2C:CONDition {START   STOP   REPEATstart   ACKMISS   ADDRess   DATA   ADDRANDDATA   ? }	
Parameter	START	Set Start as the I <sup>2</sup> C trigger condition.
	STOP	Set Stop as the I <sup>2</sup> C trigger condition.
	REPEATstart	Set Repeat of Start as the I <sup>2</sup> C trigger condition.
	ACKMISS	Set Missing Acknowledgement as the I <sup>2</sup> C trigger condition.
	ADDRess	Set Address as the I <sup>2</sup> C trigger condition.
	DATA	Set Data as the I <sup>2</sup> C trigger condition.
	ADDRANDDATA	Set Address and Data as the I <sup>2</sup> 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<sup>2</sup>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 (Set) →  
→ (Query)

Description Sets the I<sup>2</sup>C bus address type, or queries what the setting is.

Syntax :TRIGger:BUS:B1:I2C:ADDRess:TYPE {GENeralcall | STARtbyte | HSmode | EEPROM | CBUS | ? }

Related commands :TRIGger:BUS:B1:I2C:CONDition

Parameter	GENeralcall	Set a general call address (0000 000 0).
	STARtbyte	Set a start byte address. (0000 000 1)

HSmode	Set a high-speed mode address. (0000 1xx x)
EEPROM	Set an EEPROM address. (1010 xxx x)
CBUS	Set a CBUS address. (0000 001 x)

Return Parameter Returns the address type

Example :TRIGger:BUS:B1:I2C:ADDRess:TYPE?  
CBUS

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

Description Sets or queries the I<sup>2</sup>C bus address value when the I<sup>2</sup>C 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	<string>	7/10 characters, must be enclosed in double quotes, "string".  x = don't care 1 = binary 1 0 = binary 0
-----------	----------	---

Return Parameter Returns the address value.


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:ADDRes:DIRection Set →  
→ Query

Description	Sets or queries the address bit as read write or don't care.	
Note	This setting only applies when the I <sup>2</sup> C trigger is set to trigger on Address or Address/Data	
Syntax	:TRIGger:BUS:B1:I2C:ADDRes:DIRection { READ   WRITE   NOCARE   ? }	
Related commands	:TRIGger:BUS:B1:I2C:CONDition	
Parameter	READ	Set read as the data direction.
	WRITE	Set write as the data direction.
	NOCARE	Set either as the data direction.
Return Parameter	Returns the direction (READ, WRITE, NOCARE).	
Example	:TRIGger:BUS:B1:I2C:ADDRes: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 <sup>2</sup> C bus.	
 Note	This setting only applies when the I <sup>2</sup> 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<sup>2</sup>C bus when the I<sup>2</sup>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

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

**Return Parameter** Returns the data value.

**Example 1** :TRIGger:BUS:B1: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 (Set) →  
→ (Query)

**Description** Sets or queries the UART triggering condition.

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

<b>Parameter</b>	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 (Set) →  
→ (Query)

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

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

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

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


**Return Parameter** Returns the data value.

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

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

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

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

 **Note** This setting only applies when the UART trigger is set to trigger on 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.

Set →

→ Query

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

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

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

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

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

Return Parameter Returns the data value.


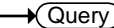
Example1 :TRIGger:BUS:B1:UART:CONDition TXDATA  
:TRIGger:BUS:B1:UART:TX:DATA:SIze 1  
:TRIGger:BUS:B1:UART:TX:DATA:VALue "1x1x0101"  
Sets the value to 1x1x0101


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


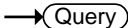
**:TRIGger:BUS:B1:SPI:CONDition** 


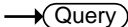
  


Description	Sets or queries the SPI triggering condition.	
Syntax	:TRIGger:BUS:B1:SPI:CONDition {SS   MISO   MOSI   MISOMOSI   ? }	
Parameter	SS	Set to trigger on the Slave Selection condition.
	MISO	Set to trigger on the Master-In Slave-Out condition.
	MOSI	Set to trigger on the Master-Out Slave-In condition.
	MISOMOSI	Set to trigger on the Master-In Slave-Out and Master-Out Slave-In conditions.
Return Parameter	Returns the triggering condition.	
Example	:TRIGger:BUS:B1:SPI:CONDition MISO Sets the SPI bus to trigger on MISO.	

**: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.	

		 
<b>:TRIGger:BUS:B1:SPI:DATa:MISO:VALue</b>		
Description	Sets or queries the triggering data value for the SPI bus when the bus is set to trigger on MISO or MISO/MOSI.	
Syntax	:TRIGger:BUS:B1:SPI:DATa:MISO:VALue {<string>   ? }	
Related commands	:TRIGger:BUS:B1:SPI:DATa:SIZE	
Parameter	<sting>	The number of characters in the string depends on the data size setting. The string must be enclosed in double quotes, "string". x = don't care 1 = binary 1 0 = binary 0
Return Parameter	Returns the data value.	
Example1	:TRIGger:BUS:B1:SPI:CONDition MISO :TRIGger:BUS:B1:SPI:DATa:SIZE 2 :TRIGger:BUS:B1:SPI:DATa:MISO:VALue "1x1x0101" Sets the value to 1x1x0101	
Example 2	:TRIGger:BUS:B1:SPI:DATa:MISO:VALue? 1X1X0101	

		 
<b>:TRIGger:BUS:B1:SPI:DATa:MOSI:VALue</b>		
Description	Sets or queries the triggering data value for the SPI bus when the bus is set to trigger on MOSI or MISO/MOSI.	
Syntax	:TRIGger:BUS:B1:SPI:DATa:MOSI:VALue {<string>   ? }	

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

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

Return Parameter Returns the data value.

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

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

Set →

← Query

**:TRIGger:BUS:B1:CAN:CONDition**

Description Sets or returns the CAN trigger condition.

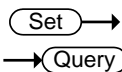
Syntax :TRIGger:BUS:B1:CAN:CONDition  
 {SOF|FRAMEtype|Identifier|DATA|IDANDDATA|EOF|ACKMISS|STUFFERR|?}

Parameter/ Return parameter	SOF	Triggers on a start of frame
	FRAMEtype	Triggers on the type of frame
	Identifier	Triggers on a matching identifier
	DATA	Triggers on matching data
	IDANDDATA	Triggers on matching identifier and data field
	EOF	Triggers on the end of frame
	ACKMISS	Triggers on a missing acknowledge

**STUFFERR** Triggers on a bit stuffing error

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

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



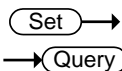
**:TRIGger:BUS:B1:CAN:FRAMeType**

Description Sets or returns the frame type for a CAN FRAMeType trigger.

Syntax :TRIGger:BUS:B1:CAN:FRAMeType  
{DATA|REMOte|ERRor|OVERLoad|?}

Parameter/ Return parameter	DATA	Sets the frame type to data frame
	REMOte	Sets the frame type to remote frame
	ERRor	Sets the frame type to error frame
	OVERLoad	Sets the frame type to overload

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



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

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

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

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

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

Set →

→ Query

**: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

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

**Example**

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

Set →

→ Query


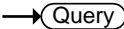
**:TRIGger:BUS:B1:CAN:IDentifier:DIRection**

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


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

<b>Parameter/ Return parameter</b>	READ	Sets read as the data direction
	WRITE	Sets write as the data direction

	<b>NOCARE</b> 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 



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|M  
OREEQUAL|?}


Parameter/ Return parameter	<b>LESSthan</b>	Triggers when the data is less than the qualifier value.
	<b>MOREthan</b>	Triggers when the data is greater than the qualifier value.
	<b>EQUAL</b>	Triggers when the data is equal to the qualifier value.
	<b>UNEQUAL</b>	Triggers when the data is not equal to the qualifier value.
	<b>LESSEQUAL</b>	Triggers when the data is less than or equal to the qualifier value.
	<b>MOREEQUAL</b>	Triggers when the data is more than or equal to the qualifier value.

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

:TRIGger:BUS:B1:CAN:DATA:SIZE (Set) →  
→ (Query)

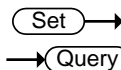
Description	Sets or returns the length of the data string in bytes for a CAN trigger.	
 Note	Only applicable when the condition is set to DATA or IDANDDATA.	
Syntax	:TRIGger:BUS:B1:CAN:DATA:SIZE {<NR1> ?}	
Parameter/ Return parameter	<NR1>	1~8 (bytes)
Example	:TRIGger:BUS:B1:CAN:DATA:SIZE? >1 :TRIGger:BUS:B1:CAN:DATA:SIZE 2 :TRIGger:BUS:B1:CAN:DATA:SIZE? >2	

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

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
```




**:TRIGger:BUS:B1:LIN:CONDition**


Description	Sets or returns the LIN trigger condition.	
Syntax	:TRIGger:BUS:B1:LIN:CONDition {SYNCField IDentifier DATA IDANDDATA WAKEup SLEEP ERRor ?}	
Parameter/ Return parameter	SYNCField	Sets the LIN trigger condition to the sync field.
	IDentifier	Sets the LIN trigger condition to identifier field.
	DATA	Sets the LIN trigger condition to the data field.
	IDANDDATA	Sets the LIN trigger condition to identifier and data field
	WAKEup	Sets the LIN trigger condition to wake up.
	SLEEP	Sets the LIN trigger condition to sleep.
	ERRor	Sets the LIN trigger condition to error.

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

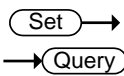
:TRIGger:BUS:B1:LIN:DATA:QUALifier (Set) →  
→ (Query)

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	

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


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

Syntax	:TRIGger:BUS:B1:LIN:DATA:SIZE {<NR1> ?}	
Parameter/ Return parameter	<NR1>	1~8 (bytes)
Example	:TRIGger:BUS:B1:LIN:DATA:SIZE? >1 :TRIGger:BUS:B1:LIN:DATA:SIZE 2 :TRIGger:BUS:B1:LIN:DATA:SIZE? >2	



**:TRIGger:BUS:B1:LIN:DATA:VALue**

**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	<pre>:TRIGger:BUS:B1:LIN:ERRTYPE? &gt;SYNC :TRIGger:BUS:B1:LIN:ERRTYPE CHECKSUM :TRIGger:BUS:B1:LIN:ERRTYPE? &gt;CHECKSUM</pre>	

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
              :TRIGger:LOGic:INPut:CLOCK:SOURce
    
```

```

:TRIGger:BUS:B1:PARAllel:VALue
    
```

→ Set  
← Query

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

**Syntax**            :TRIGger:BUS:B1:PARAllel:VALue {string}  
                       :TRIGger:BUS:B1:PARAllel:VALue?

**Related Commands**    :BUS1:PARAllel:WIDth

**Parameter**        <string>      String contents:  
   x = don't care  
   1 = binary 1  
   0 = binary 0

```

Example      :BUS1:PARAllel:WIDth 8
              :TRIGger:BUS:B1:PARAllel:VALue "1X1X0101"
              :TRIGger:BUS:B1:PARAllel:VALue?
              >1X1X0101
    
```

```

:TRIGger:LOGic:INPut:CLOCK:SOURce
    
```

→ Set  
← Query

**Description**      Sets or returns which digital channel is used as the clock source for the logic trigger. If none are selected, a pattern trigger will have to be used.

**Syntax**            :TRIGger:LOGic:INPut:CLOCK:SOURce {NONE | D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | D8 | D9 | D10 | D11 | D12 | D13 | D14 | D15 | ?}

Parameter/ Return parameter	None	No clock source selected. A pattern trigger will have to be set.
	D0~D15	Sets one of the digital channels as the clock source.

Example :TRIG:LOG:INP:CLOCK:SOUR D0  
:TRIG:LOG:INP:CLOCK:SOUR?  
>D0

**:TRIGger:LOGic:INPut:CLOCK:EDGe** (Set) →

Description Sets the polarity of the clock source.

Syntax :TRIGger:LOGic:INPut:CLOCK:EDGe {RISe | FALL | EITher}

Related Commands :TRIGger:LOGic:INPut:CLOCK:SOURce

Parameter	RISe	Sets the clock source on the rising edge.
	FALL	Sets the clock source on the falling edge.
	EITher	Sets the clock source to either rising or falling edge.

Example :TRIG:LOG:INP:CLOCK:EDG RIS

(Set) →

**:TRIGger:LOGic:FUNCTION** → (Query)

Description Sets or queries the logical combination of the digital channels for the logic trigger.

Syntax :TRIGger:LOGic:FUNCTION{AND | NAND | NOR | OR | ?}

Related Commands :TRIGger:LOGic:PATtern:INPut:D<x>

Parameter/ Return parameter	AND	Sets the logic combination to AND.
	NAND	Sets the logic combination to NAND.
	NOR	Sets the logic combination to NOR.

OR Sets the logic combination to OR.

Example :TRIGger:LOGic:FUNctioN?<br><br>>AND<br><br>:TRIGger:LOGic:FUNctioN NAND<br><br>Sets the logic combination of the digital channels to NAND.

:TRIGger:LOGic:PATtern → Query

Description Queries the conditions used for generating a logic pattern trigger in terms of input pattern, pattern trigger time and conditions.

Syntax :TRIGger:LOGic:PATtern?

Example :TIGger:LOGic:PATtern?<br><br>>:TRIGGER:LOGIC:PATTERN:INPUT:D0 HIGH; D1 X; D2 X; D3 X; D4 X; D5 X; D6 X; D7 X; D8 X; D9 X; D10 X; D11 X; D12 X; D13 X; D14 X; D15 X;<br><br>:TRIGGER:LOGIC:PATTERN:WHEN TRUE;<br><br>:TRIGGER:LOGIC:PATTERN:DELTATIME 1.000e-08;

Set →

:TRIGger:LOGic:PATtern:INPut:D<x> → Query

Description Sets or returns the logic level for the selected digital channel.

Syntax :TRIGger:LOGic:PATtern:INPut:D<x> {HIGH | LOW | X | ?}

Related Commands :TRIGger:LOGic:FUNctioN

Parameter/ Return parameter	<x>	Digital channel number 0~15.
	HIGH	Sets to logical high state.
	LOW	Sets to logical low state.
	X	Sets to "don't care" state.

Example :TRIGger:LOGic:PATtern:INPut:D0?  
>HIGH

Set →

:TRIGger:LOGic:PATtern:DELTatime

→ Query

Description Sets or returns the pattern trigger delta time value.

Syntax :TRIGger:LOGic:PATtern:DELTatime {<NR3> | ?}

Related Commands :TRIGger:LOGic:PATtern:WHEn

Parameter/ Return parameter	<NR3>	Pattern trigger time value in seconds 1e-9 (1 ns) to 10.0e0 (10 s).
--------------------------------	-------	--

Example :TRIG:LOG:PAT:DELT 8.960e-05  
:TRIG:LOG:PAT:DELT?  
>8.960e-05

Set →

:TRIGger:LOGic:PATtern:WHEn

→ Query

Description Sets or returns the pattern logic condition on which to trigger the oscilloscope.

Syntax :TRIGger:LOGic:PATtern:WHEn {TRUE | FALSE | LESSthan | MOREthan | EQUAL | UNEQual | ?}

Related Commands :TRIGger:LOGic:PATtern:DELTatime

Parameter/ Return parameter	TRUE	Triggers when the defined input pattern is met.
	FALSE	Triggers when the defined input pattern is not met.
	LESSthan	Triggers when the defined input pattern is met during a time lower than the defined delta time.
	MOREthan	Triggers when the defined input pattern is met during a time greater than the defined delta time.



---

EQUAL	Triggers when the defined input pattern is met during a time equal to the defined delta time.
UNEQUAL	Triggers when the defined input pattern is met during a time other than the defined delta time.

---

Example

```
:TRIG:LOG:PAT:DELT FALSE
:TRIG:LOG:PAT:DELT?
>FALSE
```

## System Commands

:SYSTem:LOCK ..... 186

:SYSTem:ERRor ..... 186

### :SYSTem:LOCK

Set →

→ Query

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

Set →

→ Query

Description	Queries the error queue. See the appendix on page 423 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 .....	187
:RECALL:WAVEform .....	187
:SAVe:IMAGe .....	188
:SAVe:IMAGe:FILEFormat .....	188
:SAVe:IMAGe:INKSaver .....	189
:SAVe:SETUp .....	189
:SAVe:WAVEform .....	190
:SAVe:WAVEform:FILEFormat .....	191

### :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</p> <p>Recalls setup setting S1 from memory.</p> <p>:RECALL:SETUp "Disk:/DS0001.SET"</p> <p>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

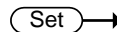
Detail CSV files cannot be recalled.

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.	

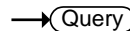
**:SAVE:IMAGe**



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.	



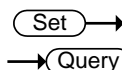
**:SAVE:IMAGe:FILEFormat**



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.	

### :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



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**



**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}

<b>Parameter</b>	CH1~REF4, CH1~CH4, Math, D0~D15, 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.)

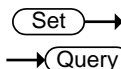
**Example 1**               :SAVe:WAVEform CH1, REF2  
 Saves the channel1 waveform to REF2.

**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**



<b>Description</b>	Sets the waveform savefile format.	
<b>Syntax</b>	:SAVe:WAVEform:FILEFormat {LSF   DCSV   FCSV   ?}	
<b>Parameter</b>	<b>LSF</b>	Sets the file format to the GDS-3000A/2000EX's internal file format, LSF. (xxx.LSF)(no support LA)
	<b>DCSV</b>	Sets the file format to detail CSV. (xxx.CSV)
	<b>FCSV</b>	Sets the file format to fast CSV. (xxx.CSV)
<b>Return parameter</b>	Returns the file format (LSF, DCSV, FCSV).	

**Example**           :SAVe:WAVEform:FILEFormat LSF  
 Sets the file format to LSF.

## Ethernet Commands

:ETHERnet:DHCP .....	192
:ETHERnet:IPADDRESS .....	192
:ETHERnet:DNS:IPADDRESS .....	193
:ETHERnet:DOMAINname .....	193
:ETHERnet:ENET:ADDRESS?.....	193
:ETHERnet:GATEWay:IPADDRESS .....	193
:ETHERnet:NAME .....	194
:ETHERnet:PASSWord.....	194
:ETHERnet:SUBNETMask.....	194

: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.	

:ETHERnet:IPADDRESS Set →  
→ Query

Description	Sets or queries ETHERnet IP address.	
Syntax	:ETHERnet:IPADDRESS{string } :ETHERnet:IPADDRESS?	
Parameter	string	IP number
Example	:ETHERnet:IPADDRESS "172.16.5.245" :ETHERnet:IPADDRESS? 172.16.5.245.	




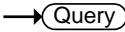

**:ETHERnet:DNS:IPADDRESS**

Description	Sets or queries ETHERnet DNS IP address.
Syntax	:ETHERnet:DNS:IPADDRESS{string } :ETHERnet:DNS:IPADDRESS?
Parameter	string IP number
Example	:ETHERnet:DNS:IPADDRESS "172.16.1.251" :ETHERnet:DNS:IPADDRESS? 172.16.1.251


**:ETHERnet:DOMAINname**

Description	Sets or queries ETHERnet domain name.
Syntax	:ETHERnet:DOMAINname {string} :ETHERnet:DOMAINname?
Parameter	string IP number
Example	:ETHERnet:DOMAINname "GW01" :ETHERnet:DOMAINname? GW01

**:ETHERnet:ENET:ADDRESS?**

Description	Queries ETHERnet MAC address.
Syntax	:ETHERnet:ENET:ADDRESS?
Example	:ETHERnet:ENET:ADDRESS? 01:02:03:04


**:ETHERnet:GATEWay:IPADDRESS**

Description	Sets or queries ETHERnet Gateway IP.
-------------	--------------------------------------

Syntax	:ETHERnet:GATEWay:IPADdRes{string} :ETHERnet:GATEWay:IPADdRes?
Parameter	string IP number
Example	:ETHERnet:GATEWay:IPADdRes "172.16.0.252" :ETHERnet:GATEWay:IPADdRes? 172.16.0.252

Set →

→ Query

**:ETHERnet:NAME**

Description	Sets or queries ETHERnet Name.
Syntax	:ETHERnet:NAME {string} :ETHERnet:NAME?
Parameter	string Instrument name
Example	:ETHERnet:NAME "GDS3000A" :ETHERnet:NAME? GDS3000A

Set →

→ Query

**:ETHERnet:PASSWord**

Description	Sets or queries ETHERnet Password.
Syntax	ETHERnet:PASSWord {string} :ETHERnet:PASSWord?
Parameter	string Password
Example	:ETHERnet:PASSWord "GW1234" :ETHERnet:PASSWord? GW1234

Set →

→ Query

**:ETHERnet:SUBNETMask**

Description	Sets or queries ETHERnet Subnet mask.
-------------	---------------------------------------

---

Syntax	ETHERnet:SUBNETMask {string} :ETHERnet:SUBNETMask?
Parameter	string Mask IP number
Example	:ETHERnet:SUBNETMask "255.255.126.0" :ETHERnet:SUBNETMask? 255.255.126.0

# Time Command

: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 "20210802142830"  
 Sets the time and date as:  
 Year: 2021, Month: 08, Day: 02, Hour: 14 (2PM),  
 Minute: 28, Second: 30.

## Bus Decode Commands

---

:BUS1 .....	198
:BUS1:STATE .....	198
:BUS1:TYPe .....	198
:BUS1:INPut.....	199
:BUS1:I2C:ADDRes:RWINClude.....	199
:BUS1:I2C:SCLK:SOURce .....	200
:BUS1:I2C:SDA:SOURce.....	200
:BUS1:PARAllel:BIT<x>:SOURce .....	200
:BUS1:PARAllel:CLOCK:EDGE .....	201
:BUS1:PARAllel:CLOCK:SOURce .....	201
:BUS1:PARAllel:WIDth .....	201
:BUS1:UART:BITRate.....	202
:BUS1:UART:DATABits.....	202
:BUS1:UART:PARlty.....	202
:BUS1:UART:PACKEt .....	203
:BUS1:UART:POLARity .....	203
:BUS1:UART:EOFPacket .....	203
:BUS1:UART:TX:SOURce .....	204
:BUS1:UART:RX:SOURce .....	204
:BUS1:SPI:SCLK:POLARity .....	205
:BUS1:SPI:SS:POLARity.....	205
:BUS1:SPI:WORDSize .....	205
:BUS1:SPI:BITORder .....	206
:BUS1:SPI:SCLK:SOURce .....	206
:BUS1:SPI:SS:SOURce.....	206
:BUS1:SPI:MOSI:SOURce .....	207
:BUS1:SPI:MISO:SOURce .....	207
:BUS1:DISPlay:FORMAt .....	208
:LISTer:DATA .....	208
:BUS1:CAN:SOURce.....	208
:BUS1:CAN:PROBe.....	209
:BUS1:CAN:SAMPLEpoint.....	209

:BUS1:CAN:BITRate .....209  
 :BUS1:LIN:BITRate .....210  
 :BUS1:LIN:IDFORmat .....210  
 :BUS1:LIN:POLARity .....211  
 :BUS1:LIN:SAMPLEpoint .....211  
 :BUS1:LIN:SOURce .....211  
 :BUS1:LIN:STANDard .....212

**:BUS1** → Query

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

Set →

**:BUS1:STATE** → Query

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

Set →

**:BUS1:TYPe** → Query

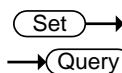
Description	Sets or queries the type of bus.
-------------	----------------------------------

Syntax :BUS1:TYPe { UART | I2C | SPI | PARAllel | CAN | LIN | ? }

Related commands :BUS1:STATE

Parameter/Return parameter	UART	Sets the bus to UART mode.
	I2C	Sets the bus to I <sup>2</sup> C mode.
	SPI	Sets the bus to SPI mode.
	PARAllel	Sets the bus to parallel mode.
	CAN	Sets the bus to CAN mode.
	LIN	Sets the bus to LIN mode.

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



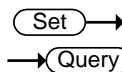
### **:BUS1:INPut**

Description Sets or returns the bus source.

Syntax :BUS1:INPut { ANAlOg | DIGItal | ? }

Parameter/Return parameter	ANAlOg	Sets the bus source as analog inputs.
	DIGItal	Sets the bus source as digital inputs.

Example1 :BUS1:INPut?  
>ANALOG



### **:BUS1:I2C:ADDRess:RWINClude**

Description Sets or queries whether the read/write bit is included in the I<sup>2</sup>C 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 I<sup>2</sup>C address.



:BUS1:I2C:SCLK:SOURce 

Description Sets or queries which channel is used for the I<sup>2</sup>C SCLK source.

Syntax :BUS1:I2C:SCLK:SOURce { CH1 | CH2 | CH3 | CH4 | D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | D8 | D9 | D10 | D11 | D12 | D13 | D14 | D15 | ? }

Parameter/Return parameter	CH1 to CH4	Analog channels 1 ~ 4.
	D0 to D15	Digital channels D0~D15

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



:BUS1:I2C:SDA:SOURce 

Description Sets or queries which channel is used for the I<sup>2</sup>C SDA source.

Syntax :BUS1:I2C:SDA:SOURce{CH1| CH2| CH3| CH4| D0| D1| D2| D3| D4| D5| D6| D7| D8| D9| D10| D11| D12| D13| D14| D15|? }

Parameter/Return parameter	CH1 to CH4	Analog channels 1 ~ 4.
	D0 to D15	Digital channels D0~D15

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



:BUS1:PARAllel:BIT<x>:SOURce 

Description Sets or returns the parallel bit source for B1.



Syntax            BUS1:PARAllel:BIT<x>:SOURce {D0| D1| D2| D3| D4| D5| D6| D7| D8| D9| D10| D11| D12| D13| D14| D15}  
 BUS1:PARAllel:BIT<x>:SOURce?

Parameter/Return <x>            the bit number  
 parameter            D0 to D15        Set the bit source B<x>

→  
 →

**:BUS1:PARAllel:CLOCK:EDGE**

Description        Sets or returns the parallel clock edge for bus1

Syntax            BUS1:PARAllel:CLOCK:EDGE{RISe| FALL|EITHer| OFF}  
 BUS1:PARAllel:CLOCK:EDGE?

Parameter        EITHer            Set either edge as the clock edge.  
                   RISe                Set the rising edge as the clock edge.  
                   FALL                Set the rising edge as the clock edge.  
                   OFF                Turn off the clock edge.

→  
 →

**:BUS1:PARAllel:CLOCK:SOURce**

Description        Sets or returns the Parallel bus1

Syntax            BUS1:PARAllel:CLOCK:SOURce{D0| D1| D2| D3| D4| D5| D6| D7| D8| D9| D10| D11| D12| D13| D14| D15}  
 BUS1:PARAllel:CLOCK:SOURce?

Parameter        D0 to D15        Set the clock source

→  
 →

**:BUS1:PARAllel:WIDth**

Description        Sets or returns the number of bits used for the width of the Parallel bus1.

Syntax            :BUS1:PARAllel:WIDth <NR1>  
 :BUS1:PARAllel:WIDth?

Parameter        <NR1>            The number of bits.

Set →

→ Query

**:BUS1:UART:BITRate**

Description	Sets or queries the UART bit rate.	
Syntax	:BUS1:UART:BITRate {<NR1>   ? }	
Parameter/Return parameter	<NR1>	UART bit rate in bps
Example	:BUS1:UART:BITRate? >2400 :BUS1:UART:BITRate 50 :BUS1:UART:BITRate? >50	

Set →

→ Query

**:BUS1:UART:DATABits**

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

Set →

→ Query

**:BUS1:UART:PARItY**

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

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

Set →

→ Query

**:BUS1:UART:PACKet**

Description Sets or queries the UART packet setting.

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

Parameter/Return parameter	<NR1>	0: Off 1: On
----------------------------	-------	-----------------

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

Set →

→ Query

**:BUS1:UART:POLARity**

Description Sets or returns the UART polarity.

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

Parameter	NORMAL	Sets normal UART polarity.
	INVerted	Sets inverted UART polarity.

Example :BUS1:UART:POLARity NORMAL  
:BUS1:UART:POLARity?  
NORMAL

Set →

→ Query

**:BUS1:UART:EOFPAcket**

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

Syntax :BUS1:UART:EOFPAcket <NR1>

Parameter/Return parameter	<NR1>	0: NULL 1: LF (line feed) 2: CR (carriage return)
----------------------------	-------	---

3: SP (space character)  
4: FF

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

Set →

:BUS1:UART:TX:SOURce

→ Query

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

Syntax :BUS1:UART:TX:SOURce { OFF | CH1 | CH2 | CH3 | CH4 | D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | D8 | D9 | D10 | D11 | D12 | D13 | D14 | D15 | ? }

Parameter/Return parameter	OFF	Off, no Tx source
	CH1 to CH4	Analog channels CH1 to CH4
	D0 to D15	Digital channels D0 to D15

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

Set →

:BUS1:UART:RX:SOURce


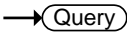
→ Query

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

Syntax :BUS1:UART:RX:SOURce { OFF | CH1 | CH2 | CH3 | CH4 | D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | D8 | D9 | D10 | D11 | D12 | D13 | D14 | D15 | ? }

Parameter/Return parameter	OFF	Off, no Rx source
	CH1 to CH4	Analog channels CH1 to CH4
	D0 to D15	Digital channels D0 to D15



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

**:BUS1:SPI:SCLK:POLARity**

---

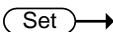
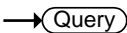
Description	Sets or queries the polarity of the SCLK line for the SPI bus.	
Syntax	:BUS1:SPI:SCLK:POLARity { FALL   RISE   ? }	
Parameter/Return parameter	FALL	Sets the polarity to falling edge.
	RISE	Sets the polarity to rising edge.
Example	:BUS1:SPI:SCLK:POLARity FALL Sets the polarity to falling edge.	

**:BUS1:SPI:SS:POLARity**

---

Description	Sets or queries the polarity of the SS line for the SPI bus.	
Syntax	:BUS1:SPI:SS:POLARity { LOW   HIGH   ? }	
Parameter/Return parameter	LOW	Active low polarity
	HIGH	Active high polarity
Example	:BUS1:SPI:SS:POLARity LOW Sets the SS line to active low.	

**:BUS1:SPI:WORDSize**

---

Description	Sets the number of bits per word for the SPI bus.	
Syntax	:BUS1:SPI:WORDSize { <NR1>   ? }	
Parameter/Return parameter	<NR1>	Bits per word (4~32)
Example	:BUS1:SPI:WORDSize 4 Sets the word size to 4 bits per word.	

**:BUS1:SPI:BITORder** 
 →  
 →

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

**:BUS1:SPI:SCLK:SOURce** 
 →  
 →

Description	Sets or queries which channel is used for the SPI SCLK source.	
Syntax	:BUS1:SPI:SCLK:SOURce { CH1   CH2   CH3   CH4   D0   D1   D2   D3   D4   D5   D6   D7   D8   D9   D10   D11   D12   D13   D14   D15   ? }	
Parameter/Return parameter	CH1 to CH4 D0 to D15	Analog channels CH1 to CH4 Digital channels D0 to D15
Example	:BUS1:SPI:SCLK:SOURce CH1 Sets channel 1 as the SPI SCLK source.	

**:BUS1:SPI:SS:SOURce** 
 →  
 →

Description	Sets or queries which channel is used for the SPI SS source.	
Syntax	:BUS1:SPI:SS:SOURce { CH1   CH2   CH3   CH4   D0   D1   D2   D3   D4   D5   D6   D7   D8   D9   D10   D11   D12   D13   D14   D15   ? }	
Parameter/Return parameter	CH1 to CH4 D0 to D15	Analog channels CH1 to CH4 Digital channels D0 to D15

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 | D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | D8 | D9 | D10 | D11 | D12 | D13 | D14 | D15 | ? }

Parameter/Return parameter	CH1 to CH4	Analog channels CH1 to CH4
	D0 to D15	Digital channels D0 to D15
	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 | D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | D8 | D9 | D10 | D11 | D12 | D13 | D14 | D15 | ? }

Parameter/Return parameter	CH1 to CH4	Analog channels CH1 to CH4
	D0 to D15	Digital channels D0 to D15
	OFF	No MISO source.

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

**:BUS1:DISPlay:FORMAt** (Set) →  
→ (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 HEXadecimal	Binary format 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.

**:BUS1:CAN:SOURce** (Set) →  
→ (Query)

Description	Sets or returns the CAN input source.	
Syntax	:BUS1:CAN:SOURce { CH1   CH2   CH3   CH4   D0   D1   D2   D3   D4   D5   D6   D7   D8   D9   D10   D11   D12   D13   D14   D15   ? }	
Parameter/Return parameter	CH1 ~ CH4 D0 to D15	Analog channel source Digital channels D0 to D15
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   <NR1>   ?}

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

Example

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

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

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

:BUS1:LIN:BITRate Set →  
→ Query

Description	Sets or returns the bit rate of the LIN bus.	
Syntax	:BUS1:LIN:BITRate {<NR1>   ?}	
Parameter/Return parameter	<NR1>	LIN bit rate in bps.

Example

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

:BUS1:LIN:IDFORmat Set →  
→ Query

Description	Sets or returns the LIN ID format.	
Syntax	:BUS1:LIN:IDFORmat {NOPARity PARity ?}	

Parameter/Return parameter	NOPARity	Don't include parity bits with Id.
	PARity	Include parity bits with Id.
Example	:BUS1:LIN:IDFORmat? NOPARITY Returns the ID format.	

Set →  
→ Query

**:BUS1:LIN:POLARity**

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

**:BUS1:LIN:SAMPLEpoint**

→ Query

Description	Returns the sample point.	
Syntax	:BUS1:LIN:SAMPLEpoint?	
Return Parameter	Returns the sample point of the LIN bus as a percentage.	
Example	:BUS1:LIN:SAMPLEpoint? 50 Returns the sample point as a percentage.	

Set →  
→ Query

**:BUS1:LIN:SOURce**

Description	Sets or returns the LIN data source.	
-------------	--------------------------------------	--

Syntax :BUS1:LIN:SOURce {CH1 | CH2 | CH3 | CH4 | D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | D8 | D9 | D10 | D11 | D12 | D13 | D14 | D15 | ? }

Parameter/Return parameter	CH1 ~ CH4	Analog channel source
	D0 to D15	Digital channels D0 to D15

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

Set →

→ Query

**: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 :BUS1:LIN:STANDARD?  
>BOTH  
Returns the LIN standard.

## Mark Commands

:MARK .....	213
:MARK:CREATE.....	213
:MARK:DELEte .....	214

### :MARK Set →

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 Set →

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.	

## Mask Commands

---

:MASK:STATe.....	216
:MASK:SOURce .....	216
:MASK:VIOLation .....	216
:MASK:VIOLation:SAVe.....	217
:MASK:AUTO.....	217
:MASK:AUTO:SOURce .....	218
:MASK:AUTO:UNITs .....	218
:MASK:USER:UNITs .....	218
:MASK:USER:AREa<x1>:POINt<x2>.....	219
:MASK:USER:AREa<x1>:POINt<x2>:STATe .....	219
:MASK:USER:SAVe .....	220
:MASK:USER:LOAD .....	220
:MASK:USER:CREATE .....	220
:MASK:RATIO .....	221
:MASK:VIOLation:THREShold.....	221
:MASK:STOPAFter .....	221
:MASK:STOPAFter:WAVEform .....	222
:MASK:STOPAFter:TIME .....	222
:MASK:FAILACTion:STOPACQuisition .....	223
:MASK:FAILACTion:SAVEIMAge.....	223
:MASK:FAILACTion:SAVEWAVEform .....	223
:MASK:FAILACTion:HARDCopy.....	224
:MASK:FAILACTion:GONogo.....	224
:MASK:COMPLETEACTion:GONogo.....	225
:MASK:COMPLETEACTion:TESTREpeat.....	225
:MASK:COMPLETEACTion:TESTDelay .....	226
:MASK:RESults:DISPlay .....	226
:MASK:RESults:DISPlay:DETail .....	226
:MASK:RESults:CURRent:STATUS .....	227
:MASK:RESults:CURRent:WAVEform .....	227
:MASK:RESults:CURRent:VIOLation.....	227
:MASK:RESults:CURRent:TIME .....	228

:MASK:RESults:CURRent:HITSPERSEGment<x>.....228  
 :MASK:RESults:TOTal:WAVEform .....228  
 :MASK:RESults:TOTal:VIOLation.....228  
 :MASK:RESults:TOTal:TESTNUMber .....229  
 :MASK:RESults:TOTal:FTTESTNUMber .....229  
 :MASK:RESults:TOTal:TIME.....229  
 :MASK:RESults:TOTal:HITSPERSEGment<x>.....229

**:MASK:STATe** 



Description	Sets the mask state to on or off. Or returns the mask state.	
Syntax	:MASK:STATe {ON OFF} :MASK:STATe?	
Parameter	ON	Turn the mask function on.
	OFF	Turn the mask function off.
Example	:MASK:STATe ON Turn mask on.	

**:MASK:SOURce** 



Description	Sets or returns the compared source.	
Syntax	:MASK:SOURce {CH1 CH2 CH3 CH4} :MASK:SOURce?	
Parameter	CH1~CH4:	Channel 1 to Channel 4.
Example	:MASK:SOURce CH1 Set the compared source as channel 1.	

**:MASK:VIOLation** 

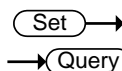


Description	Set or returns actions for the mask violations.	
-------------	---	--



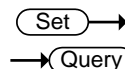
Syntax	:MASK:VIOLation {STOP CONTInue} :MASK:VIOLation?	
Parameter	STOP	The waveform will be frozen.
	CONTInue	Ignore the violation.
Example	:MASK:VIOLation STOP Sets the violation action to stop.	

**:MASK:VIOLation:SAVE**



Description	Turns on/off the function of saving violation images. Or returns the state of saving violation images function.	
Syntax	:MASK:VIOLation:SAVE {ON OFF} :MASK:VIOLation:SAVE?	
Parameter	ON	Turns on this function.
	OFF	Turns off this function.
Example	:MASK:VIOLation:SAVE ON Turns the function of saving violation images on.	

**:MASK:AUTO**



Description	Creates a mask fast in according to the reference source. Or returns the setting of the mask.	
Syntax	:MASK:AUTO {<Xmask>,<Ymask>} :MASK:AUTO?	
Parameter	<Xmask>	Sets the horizontal range for the mask.
	<Ymask>	Sets the vertical range for the mask.
Example	:MASK:AUTO 0.2,1.2 Sets the range of mask as {0.2,1.2} and creates a mask.	

**:MASK:AUTO:SOURce** 


Description	Sets or returns the reference source for the mask.	
Syntax	:MASK:AUTO:SOURce {CH1 CH2 CH3 CH4} :MASK:AUTO:SOURce?	
Parameter	CH1~CH4	Channel 1 to Channel 4.
Example	:MASK:AUTO:SOURce CH1 Sets the reference source for the mask as channel 1.	

**:MASK:AUTO:UNITs** 


Description	Sets or returns the units for auto created mask.	
Syntax	:MASK:AUTO:UNITs {DIVisions CURRent} :MASK:AUTO:UNITs?	
Parameter	DIVisions	Divisions units.
	CURRent	Current units.
Example	:MASK:AUTO:UNITs DIVisions Sets the units for auto created mask as divisions units.	

**:MASK:USER:UNITs** 


Description	Sets or returns the units for the customized mask.	
Syntax	:MASK:USER:UNITs {DIVisions CURRent} :MASK:USER:UNITs?	
Parameter	DIVisions	Divisions units.
	CURRent	Current units.

**Example** :MASK:USER:UNITs DIVisions  
 Sets the units for the customized mask as divisions units.

:MASK:USER:AREa<x1>:POINT<x2>  

**Description** Sets or returns the coordinates for the customized mask.

**Syntax** :MASK:USER:AREa<x1>:POINT<x2>  
 {<XMASK>,<YMASK>}  
 :MASK:USER:AREa<x1>:POINT<x2>?

<b>Parameter</b>	<b>&lt;x1&gt;</b>	Number of the customized mask. <x1>:1~8.
	<b>&lt;x2&gt;</b>	Number of the points set up the mask area. <x2>:1~10.
	<b>&lt;XMASK&gt;</b>	Horizontal coordinates.<NRF>
	<b>&lt;YMASK&gt;</b>	Vertical coordinates.<NRF>

**Example** :MASK:USER:AREa1:POINT1 2,2  
 Sets point 1 as (2,2) in area 1.

:MASK:USER:AREa<x1>:POINT<x2>:STATe  

**Description** Sets or returns the state of the point in the mask area.

**Syntax** :MASK:USER:AREa<x1>:POINT<x2>:STATe {ON|OFF}  
 :MASK:USER:AREa<x1>:POINT<x2>:STATe?

<b>Parameter</b>	<b>AREa&lt;x1&gt;</b>	Number of the customized mask. <x1>:1~8.
	<b>POINT&lt;x2&gt;</b>	Number of the points set up the mask area. <x2>:1~10.
	<b>ON</b>	Enable the point in the mask area.
	<b>OFF</b>	Disable the point in the mask area.

**Example** :MASK:USER:AREa1:POINt1:STATe ON  
 Set the state of the point 1 in area 1 to ON.

**:MASK:USER:SAVE** (Set) →

**Description** Saves a customized mask to the assigned file path with a specified filename.

**Syntax** :MASK:USER:SAVE <file path>  
 ("Disk:/xxx.MSK", "USB:/xxx.MSK")

**Parameter** xxx.MSK      Filename.

**Example** :MASK:USER:SAVE "Disk:/mask1.MSK"  
 Saves a customized mask named mask1.MSK to root directory (Disk:/) of the scope.

**:MASK:USER:LOAD** (Set) →

**Description** Loads a customized mask from the assigned file path with a specified filename.

**Syntax** :MASK:USER:LOAD <file path>  
 ("Disk:/xxx.MSK", "USB:/xxx.MSK")

**Parameter** xxx.MSK      Filename

**Example** :MASK:USER:LOAD "Disk:/mask1.MSK"  
 Loads a customized mask named mask1.MSK from root directory (Disk:/) of the scope.

**:MASK:USER:CREATE** (Set) →

**Description** Create or removes the customized mask.

**Syntax** :MASK:USER:CREATE {ON|OFF}

**Parameter** ON              Creates the mask.  
 OFF                      Removes the mask.

Example :MASK:USER:CREATE ON  
Create a customized mask.

Set →

→ Query

:MASK:RATio

Description Resets the mask violation ratio.

Syntax :MASK:RATio {RESET}  
:MASK:RATio?

Parameter RESET Reset

Example :MASK:RATio RESET  
Resets the ratio.

Set →

→ Query

:MASK:VIOLation:THREShold

Description Sets or queries the violation threshold for mask.

Syntax :MASK:VIOLation:THREShold {<NR1> | INFInite}  
:MASK:VIOLation:THREShold?

Parameter <NR1> Sets the violation threshold.  
Range:1~1000000  
INFInite Sets violation threshold as infinite.

Example :MASK:VIOLation:THREShold 1000  
:MASK:VIOLation:THREShold?  
1000

Set →

→ Query

:MASK:STOPAFter

Description Sets or queries the mask test stop after waveform or time violation.

Syntax :MASK:STOPAFter {WAVEform | TIME}  
:MASK:STOPAFter?

Parameter WAVEform Sets the mask test stop after waveform violation.

**TIME** Sets the mask test stop after time violation.

Example :MASK:STOPAFter TIME  
 :MASK:STOPAFter?  
 TIME

Set →

→ Query

:MASK:STOPAFter:WAVeform

Description Sets or queries the numbers of waveform for stop test.

Syntax :MASK:STOPAFter:WAVeform {<NR1> | INFINite}  
 :MASK:STOPAFter:WAVeform?

Parameter <NR1> Sets the numbers of waveform for stop test. Range:1~1000000  
 INFINite Sets the numbers of waveform for stop test as infinite.

Example :MASK:STOPAFter:WAVeform 20000  
 :MASK:STOPAFter:WAVeform?  
 20000

Set →

→ Query

:MASK:STOPAFter:TIME

Description Sets or queries the numbers of time for stop test.

Syntax :MASK:STOPAFter:TIME{<NR1> | INFINite}  
 :MASK:STOPAFter:TIME?

Parameter <NR1> Sets the numbers of time for stop test. Range:1~1000000  
 INFINite Sets the numbers of time for stop test as infinite.

Example :MASK:STOPAFter:TIME 20000  
 :MASK:STOPAFter:TIME?  
 20000

**:MASK:FAILACTion:STOPACQuisition** 


Description	Sets or queries the state of action, stop the test when test fail.	
Syntax	:MASK:FAILACTion:STOPACQuisition {ON OFF} :MASK:FAILACTion:STOPACQuisition?	
Parameter	ON	Turns on the fail action: stop acuquisition.
	OFF	Turns off the fail action: stop acuquisition.
Example	:MASK:FAILACTion:STOPACQuisition ON :MASK:FAILACTion:STOPACQuisition? ON	

**:MASK:FAILACTion:SAVEIMAge** 


Description	Sets or queries the state of action, saving the screen image when test fail.	
Syntax	:MASK:FAILACTion:SAVEIMAge {ON OFF} :MASK:FAILACTion:SAVEIMAge?	
Parameter	ON	Turns on the fail action: saving the screen image.
	OFF	Turns off the fail action: saving the screen image.
Example	:MASK:FAILACTion:SAVEIMAge ON :MASK:FAILACTion:SAVEIMAge? ON	

**:MASK:FAILACTion:SAVEWAVeform** 


Description	Sets or queries the state of action, saving the waveform file when test fail.	
-------------	---	--

Syntax :MASK:FAILACTion:SAVEWAVEform {ON|OFF}  
:MASK:FAILACTion:SAVEWAVEform?

Parameter ON Turns on the fail action: saving the waveform file.  
OFF Turns off the fail action: saving the waveform file.

Example :MASK:FAILACTion:SAVEWAVEform ON  
:MASK:FAILACTion:SAVEWAVEform?  
ON

Set →

→ Query

**:MASK:FAILACTion:HARDCopy**

Description Sets or queries the state of action, hardcopy when test fail.

Syntax :MASK:FAILACTion:HARDCopy {ON|OFF}  
:MASK:FAILACTion:HARDCopy?

Parameter ON Turns on the fail action:hardcopy.  
OFF Turns off the fail action:hardcopy.

Example :MASK:FAILACTion:HARDCopy ON  
:MASK:FAILACTion:HARDCopy?  
ON

Set →

→ Query

**:MASK:FAILACTion:GONogo**

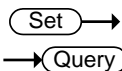
Description Sets or queries the state of action, output a signal from Go/NoGo port when test fail.

Syntax :MASK:FAILACTion:GONogo {ON|OFF}  
:MASK:FAILACTion:GONogo?

Parameter ON Turns on the fail action:output a signal from Go/NoGo port.  
OFF Turns off the fail action:output a signal from Go/NoGo port.



Example :MASK:FAILACTion:GONogo ON  
 :MASK:FAILACTion:GONogo?  
 ON



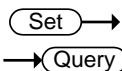
**:MASK:COMPLETEACTion:GONogo**

Description Sets or queries the state of action, output a signal from Go/NoGo port when test complete.

Syntax :MASK:COMPLETEACTion:GONogo {ON|OFF}  
 :MASK:COMPLETEACTion:GONogo?

Parameter	ON	Turns on the complete action:output a signal from Go/NoGo port.
	OFF	Turns off the complete action:output a signal from Go/NoGo port.

Example :MASK:COMPLETEACTion:GONogo ON  
 :MASK:COMPLETEACTion:GONogo?  
 ON



**:MASK:COMPLETEACTion:TESTREpeat**

Description Sets or queries the state of action, repeat the test when test complete.

Syntax :MASK:COMPLETEACTion:TESTREpeat {ON|OFF}  
 :MASK:COMPLETEACTion:TESTREpeat?

Parameter	ON	Turns on the complete action:repeat the test.
	OFF	Turns off the complete action:repeat the test.

Example :MASK:COMPLETEACTion:TESTREpeat ON  
 :MASK:COMPLETEACTion:TESTREpeat?  
 ON

Set →  
 → Query

**:MASK:COMPLETEACTion:TESTDelay**

---

Description	Sets or queries the delay time of test repeat.
Syntax	:MASK:COMPLETEACTion:TESTDelay {<NRF> :MASK:COMPLETEACTion:TESTDelay?
Parameter	<NRF> Sets the times of delay. Range:0~200(sec)
Example	:MASK:COMPLETEACTion:TESTDelay 5 :MASK:COMPLETEACTion:TESTDelay? 5.0

Set →  
 → Query

**:MASK:RESults:DISPlay**

---

Description	Sets or queries the display state of mask results menu.
Syntax	:MASK:RESults:DISPlay {ON OFF} :MASK:RESults:DISPlay?
Parameter	ON Turns on the mask results menu. OFF Turns off the mask results menu.
Example	:MASK:RESults:DISPlay ON :MASK:RESults:DISPlay? ON

Set →  
 → Query

**:MASK:RESults:DISPlay:DETail**

---

Description	Sets or queries the display state of mask detail results menu.
Syntax	:MASK:RESults:DISPlay:DETail {ON OFF} :MASK:RESults:DISPlay:DETail?
Parameter	ON Turns on the mask detail results menu. OFF Turns off the mask detail results menu.

Example :MASK:RESults:DISPlay:DETail ON  
:MASK:RESults:DISPlay:DETail?  
ON

### :MASK:RESults:CURRent:STATUS

→ Query

Description	Queries the current status of mask test.	
Syntax	:MASK:RESults:CURRent:STATUS?	
Parameter	OFF	Mask test is off.
	Running (Failing)	Mask test is running, and the test is failed.
	Running (Passing)	Mask test is running, and the test is passed.
	Failed	Mask test stopped, and the test is failed.
	Passed	Mask test stopped, and the test is passed.
	Delay	Mask test stopped, and waiting for repeat test.

Example :MASK:RESults:CURRent:STATUS?  
Passed

### :MASK:RESults:CURRent:WAVEform

→ Query

Description	Queries the current numbers of tested waveforms.	
Syntax	:MASK:RESults:CURRent:WAVEform?	
Example	:MASK:RESults:CURRent:WAVEform? 638	

### :MASK:RESults:CURRent:VIOLation

→ Query

Description	Queries the current numbers of violations.	
-------------	--	--

Syntax :MASK:RESults:CURRent:VIOLation?

Example :MASK:RESults:CURRent:VIOLation?  
176

**:MASK:RESults:CURRent:TIME** → Query

Description Queries the current elapsed time(sec).

Syntax :MASK:RESults:CURRent:TIME?

Example :MASK:RESults:CURRent:TIME?  
5

**:MASK:RESults:CURRent:HITSPERSEGment<x>** → Query

Description Queries the current detailed results of hits per segment mask.

Syntax :MASK:RESults:CURRent:HITSPERSEGment<x>?

Parameter <x> 1~8

Example :MASK:RESults:CURRent:HITSPERSEGment2?  
255

**:MASK:RESults:TOTal:WAVEform** → Query

Description Queries the total numbers of tested waveforms.

Syntax :MASK:RESults:TOTal:WAVEform?

Example :MASK:RESults:TOTal:WAVEform?  
1256

**:MASK:RESults:TOTal:VIOLation** → Query

Description Queries the total numbers of violations.

Syntax :MASK:RESults:TOTal:VIOLation?

Example :MASK:RESults:TOTal:VIOLation?  
562

:MASK:RESults:TOTal:TESTNUMBER → Query

Description Queries the total numbers of completed mask test.

Syntax :MASK:RESults:TOTal:TESTNUMBER?

Example :MASK:RESults:TOTal:TESTNUMBER?  
10

:MASK:RESults:TOTal:FTESTNUMBER → Query

Description Queries the total numbers of failed mask test.

Syntax :MASK:RESults:TOTal:FTESTNUMBER?

Example :MASK:RESults:TOTal:FTESTNUMBER?  
3

:MASK:RESults:TOTal:TIME → Query

Description Queries the total elapsed time(sec).

Syntax :MASK:RESults:TOTal:TIME?

Example :MASK:RESults:TOTal:TIME?  
150

:MASK:RESults:TOTal:HITSPERSEGment<x> → Query

Description Queries the total detailed results of hits per segment mask.

Syntax :MASK:RESults:TOTal:HITSPERSEGment<x>?

Parameter <x> 1~8

Example :MASK:RESults:TOTal:HITSPERSEGment1?  
36

## FRA Commands

:FRA:RUN.....	230
:FRA:STOP .....	231
:FRA:SOURce:INPut .....	231
:FRA:SOURce:OUTPut .....	231
:FRA:FREQuency:STARt .....	232
:FRA:FREQuency:STOP .....	232
:FRA:AWG:LOAd.....	232
:FRA:AWG:AMPlitude.....	233
:FRA:POINt .....	234
:FRA:SAVe .....	234
:FRA:RECALL.....	234
:FRA:DATA .....	235
:FRA:SAVETOCsv.....	235
:FRA:AWG:AMPlitude:PROFile .....	235
:FRA:AWG:AMPlitude:INTERPolation .....	235
:FRA:DATA:GMARgin.....	236
:FRA:DATA:GMARgin:FREQuency.....	236
:FRA:DATA:PMARgin .....	237
:FRA:DATA:PMARgin:FREQuency .....	237
:FRA:STATe .....	237

:FRA:RUN 


Description	Runs the FRA function or returns the FRA state.
Syntax	:FRA:RUN :FRA:RUN?
Example	:FRA:RUN FRA starts.

**:FRA:STOP** 



**Description** Stops the FRA function or returns the FRA state.

**Syntax** :FRA:STOP  
:FRA:STOP?

**Example** :FRA:STOP  
FRA stops.

**:FRA:SOURce:INPut** 



**Description** Sets or returns the input source for FRA.

**Syntax** :FRA:SOURce:INPut {CH1~CH4}  
:FRA:SOURce:INPut?

**Parameter** CH1~CH4 Channel 1 to Channel 4

**Example** :FRA:SOURce:INPut CH1  
Set the input source as channel 1.

**:FRA:SOURce:OUTPut** 



**Description** Sets or returns the output source for FRA.

**Syntax** :FRA:SOURce:OUTPut {CH1~CH4}  
:FRA:SOURce:OUTPut?

**Parameter** CH1~CH4 Channel 1 to Channel 4

**Example** :FRA:SOURce:OUTPut CH2  
Set the input source as channel 2.

**:FRA:FREQuency:STARt** (Set) →  
→ (Query)

Description	Sets or returns the start frequency for FRA.	
Syntax	:FRA:FREQuency:STARt {<NRf>} :FRA:FREQuency:STARt?	
Parameter	<NRf>	Sets the frequency to use. (Range:20Hz~25MHz)
Example	:FRA:FREQuency:STARt 100 Sets the start frequency as 100Hz.	

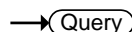
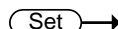
**:FRA:FREQuency:STOP** (Set) →  
→ (Query)

Description	Sets or returns the stop frequency for FRA.	
Syntax	:FRA:FREQuency:STOP {<NRf>} :FRA:FREQuency:STOP?	
Parameter	<NRf>	Sets the frequency to use. (Range:20Hz~25MHz)
Example	:FRA:FREQuency:STOP 500 Sets the start frequency as 500Hz.	

**:FRA:AWG:LOAD** (Set) →  
→ (Query)

Description	Sets or returns the impedance for load.	
Syntax	:FRA:AWG:LOAD {FIFty HIGHZ} :FRA:AWG:LOAD?	
Parameter	FIFty	50 ohm
	HIGHZ	High impedance
Example	:FRA:AWG:LOAD HIGHZ Sets the load as high impedance.	





**:FRA:AWG:AMPlitude**

**Description**            Sets or returns the amplitude for FRA.  
                                  When amplitude profile is on, sets or returns the  
                                  amplitude for the selected frequency.

**Syntax**                    :FRA:AWG:AMPlitude {<NRf>,<range>}]  
                                  :FRA:AWG:AMPlitude? [<range>]

**Parameter**            <NRf>    Sets the amplitude to user.(Load=FIFTY,  
                                  Range:0.01Vpp~2.5Vpp; Load=HIGHZ,  
                                  Range:0.02Vpp~5Vpp)  
                                  <range>    The selected frequency.  
                                  {F20hz | F100hz | F1Khz | F10Khz | F100Khz  
                                  | F1Mhz | F10Mhz | F25Mhz}  
                                  F20hz: Frequency range >20Hz (The  
                                  default <range>).  
                                  F100hz: Frequency range >100Hz.  
                                  F1Khz: Frequency range >1kHz.  
                                  F10Khz: Frequency range >10kHz.  
                                  F100Khz: Frequency range >100kHz.  
                                  F1Mhz: Frequency range >1MHz.  
                                  F10Mhz: Frequency range >10MHz.  
                                  F25Mhz: Frequency range 25MHz.

**Example**                    :FRA:AWG:AMPlitude 0.2  
                                  Sets the amplitude as 0.2Vpp.  
                                  :FRA:AWG:AMPlitude 0.5,F100HZ  
                                  :FRA:AWG:AMPlitude? F100HZ  
                                  0.5

		Set →
:FRA:POINt		→ Query
Description	Sets or returns the number of processing points in a decade.	
Syntax	:FRA:POINt {<NR1>} :FRA:POINt?	
Parameter	<NR1>	The number of points in a decade. (Range:10, 15, 30, 45, 90)
Example	:FRA:POINt 15  Sets the number of processing points as 15 in a decade.	

		Set →
:FRA:SAVe		
Description	Saves the FRA result.	
Syntax	:FRA:SAVe	
Example	:FRA:SAVe  Saves the result to default file.	

		Set →
:FRA:RECALL		
Description	Recalls the FRA result from memory or USB.	
Syntax	:FRA:RECALL {<file path> ("Disk:/xxx.FRD", "USB:/xxx.FRD")}	
Parameter	xxx.FRD	Filename
Example	:FRA:RECALL "Disk:/FRA1.FRD"  Recalls a FRA result named FRA1.FRD from root directory (Disk:/) of the scope.	

:FRA:DATA → Query

---

Description Shows the detailed information of FRA settings and results.

---

Syntax :FRA:DATA?

---

Example :FRA:DATA?  
Shows the FRA result's detail.

:FRA:SAVETOCsv Set →

---

Description Saves the FRA result as a CSV file.

---

Syntax :FRA:SAVETOCsv

---

Example :FRA: SAVETOCsv  
Saves results as CSV file.

:FRA:AWG:AMPlitude:PROFile Set →  
→ Query

---

Description Sets or returns the state of amplitude profile.

---

Syntax :FRA:AWG:AMPlitude:PROFile {ON|OFF}  
:FRA:AWG:AMPlitude:PROFile?

---

Parameter	ON	Turn on the profile.
	OFF	Turn off the profile.

---

Example :FRA:AWG:AMPlitude:PROFile ON  
:FRA:AWG:AMPlitude:PROFile?  
ON

:FRA:AWG:AMPlitude:INTERPolation Set →  
→ Query

---

Description Sets or returns the state of linear interpolation for the selected frequency.

---

Syntax	:FRA:AWG:AMPlitude:INTERPolation {ON OFF,[<range>]} :FRA:AWG:AMPlitude:INTERPolation? [<range>]	
Parameter	ON	Turn off the interpolation.
	OFF	Turn off the interpolation.
	<range>	The selected frequency. {F20hz   F100hz   F1Khz   F10Khz   F100Khz   F1Mhz   F10Mhz   F25Mhz}
		F20hz: Frequency range >20Hz(The default <range>).
		F100hz: Frequency range >100Hz.
		F1Khz: Frequency range >1kHz.
		F10Khz: Frequency range >10kHz.
		F100Khz: Frequency range >100kHz.
		F1Mhz: Frequency range >1MHz.
		F10Mhz: Frequency range >10MHz.
		F25Mhz: Frequency range 25MHz.

Example :FRA:AWG:AMPlitude:INTERPolation ON,F100HZ  
:FRA:AWG:AMPlitude:INTERPolation? F100HZ  
ON

:FRA:DATA:GMARgin → Query

Description Returns the gain margin of FRA results.

Syntax :FRA:DATA:GMARgin?

:FRA:DATA:GMARgin:FREQUency → Query

Description Returns the gain margin frequency of FRA results.

Syntax :FRA:DATA:GMARgin:FREQUency?

**:FRA:DATA:PMARgin** → Query

Description	Returns the phase margin of FRA results.
Syntax	:FRA:DATA:PMARgin?

**:FRA:DATA:PMARgin:FREQuency** → Query

Description	Returns the phase margin frequency of FRA results.
Syntax	:FRA:DATA:PMARgin:FREQuency?

**:FRA:STATe** → Query

Description	Query or turn on/off the FRA function.	
Syntax	:FRA:STATe {ON OFF} :FRA:STATe?	
Parameter	ON	Turn on the FRA. (No work when PWR function running)
	OFF	Turn off the FRA.
Example	:FRA:STATe ON :FRA:STATe? ON	

## Search Commands

---

:SEARCH:COPIY .....	240
:SEARCH:STATE .....	240
:SEARCH:TOTAL .....	240
:SEARCH:TRIGger:TYPe .....	241
:SEARCH:TRIGger:SOURce.....	241
:SEARCH:TRIGger:EDGE:SLOP .....	241
:SEARCH:TRIGger:LEVel .....	242
:SEARCH:TRIGger:HLEVel .....	243
:SEARCH:TRIGger:LLEVel .....	243
:SEARCH:TRIGger:PULSEWidth:POLarity .....	243
:SEARCH:TRIGger:RUNT:POLarity .....	244
:SEARCH:TRIGger:RISEFall:SLOP .....	244
:SEARCH:TRIGger:PULSE:WHEn.....	245
:SEARCH:TRIGger:PULSE:TiMe.....	245
:SEARCH:TRIGger:RUNT:WHEn .....	246
:SEARCH:TRIGger:RUNT:TiMe .....	246
:SEARCH:TRIGger:RISEFall:WHEn .....	247
:SEARCH:TRIGger:RISEFall:TiMe .....	247
:SEARCH:TRIGger:BUS:TYPe.....	248
:SEARCH:TRIGger:BUS:B1:I2C:CONDition .....	248
:SEARCH:TRIGger:BUS:B1:I2C:ADDReSS:MODE .....	249
:SEARCH:TRIGger:BUS:B1:I2C:ADDReSS:TYPe .....	249
:SEARCH:TRIGger:BUS:B1:I2C:ADDReSS:VALue.....	250
:SEARCH:TRIGger:BUS:B1:I2C:ADDReSS	
:DIRection .....	251
:SEARCH:TRIGger:BUS:B1:I2C:DATA:SIZE .....	251
:SEARCH:TRIGger:BUS:B1:I2C:DATA:VALue .....	252
:SEARCH:TRIGger:BUS:B1:UART:CONDition.....	253
:SEARCH:TRIGger:BUS:B1:UART:RX:DATA:SIZE.....	253
:SEARCH:TRIGger:BUS:B1:UART:RX:DATA:VALue .....	254
:SEARCH:TRIGger:BUS:B1:UART:TX:DATA:SIZE .....	256
:SEARCH:TRIGger:BUS:B1:UART:TX:DATA:VALue .....	256
:SEARCH:TRIGger:BUS:B1:SPI:CONDition.....	257

:SEARCH:TRIGger:BUS:B1:SPI:DATa:SIZE.....	257
:SEARCH:TRIGger:BUS:B1:SPI:DATa:MISO:VALue	258
:SEARCH:TRIGger:BUS:B1:SPI:DATa:MOSI:VALue	259
:SEARCH:TRIGger:BUS:B1:CAN:CONDition .....	259
:SEARCH:TRIGger:BUS:B1:CAN:FRAMeType .....	260
:SEARCH:TRIGger:BUS:B1:CAN:IDentifier:MODe	261
:SEARCH:TRIGger:BUS:B1:CAN:IDentifier:VALue	261
:SEARCH:TRIGger:BUS:B1:CAN:IDentifier	
:DIRection .....	262
:SEARCH:TRIGger:BUS:B1:CAN:DATa:QUALifier...	263
:SEARCH:TRIGger:BUS:B1:CAN:DATa:SIZE .....	264
:SEARCH:TRIGger:BUS:B1:CAN:DATa:VALue .....	264
:SEARCH:TRIGger:BUS:B1:LIN:CONDition .....	265
:SEARCH:TRIGger:BUS:B1:LIN:DATa:QUALifier ....	266
:SEARCH:TRIGger:BUS:B1:LIN:DATa:SIZE .....	267
:SEARCH:TRIGger:BUS:B1:LIN:DATa:VALue .....	267
:SEARCH:TRIGger:BUS:B1:LIN:ERRTYpe .....	268
:SEARCH:TRIGger:BUS:B1:LIN:IDentifier:VALue....	268
:SEARCH:FFTPeak:METhod .....	269
:SEARCH:FFTPeak:METhod:MPEak.....	270
:SEARCH:FFTPeak:SINFo.....	270
:SEARCH:FFTPeak:LIST .....	271
:SEARCH:FFTPeak:SOURce .....	271
:SEARCH:TRIGger:LOGic:INPut:CLOCK:SOURce ..272	
:SEARCH:TRIGger:LOGic:INPut:CLOCK:EDGe .....	272
:SEARCH:TRIGger:LOGic:FUNCTion .....	273
:SEARCH:TRIGger:LOGic:PATtern.....	273
:SEARCH:TRIGger:LOGic:PATtern:INPut:D<x> .....	274
:SEARCH:TRIGger:LOGic:PATtern:DELtatime .....	274
:SEARCH:TRIGger:LOGic:PATtern:WHEn .....	275
:SEARCH:TRIGger:BUS:TYpe .....	275

**:SEARCH:COPY**

Set →

**Description** Copies the search settings to the trigger settings or copies the trigger settings to the search settings.

**Syntax** :SEARCH:COPY {SEARCHtotrigger|TRIGgertosearch}

<b>Parameter</b>	SEARCHtotrigger	Copy the search setting to the trigger settings.
	TRIGgertosearch	Copy the trigger settings to the search settings.

**Example** :SEARCH:COPY SEARCHtotrigger  
Copies the search settings to the trigger settings.

Set →

**:SEARCH:STATE**

→ Query

**Description** Sets or queries whether the Search function is on or off.

**Syntax** :SEARCH:STATE { OFF | ON | ? }

<b>Parameter/Return parameter</b>	OFF	Turn the Search function on.
	ON	Turn the Search function off.

**Example** :SEARCH:STATE ON  
Turn Search on.

**:SEARCH:TOTAL**

→ Query

**Description** Returns the total number of events found from the search function.

**Syntax** :SEARCH:TOTAL?

**Return parameter** <NR1> Number of events.

**Example** :SEARCH:TOTAL?  
5



**:SEARCH:TRIGger:TYPe** 
 →  
 →

Description	Sets or queries the search trigger type.	
Syntax	:SEARCH:TRIGger:TYPe { EDGe   PULSEWidth   RUNT   RISEFall   FFTPeak   LOGic   BUS   ? }	
Parameter/Return parameter	EDGe	Edge trigger
	PULSEWidth	Pulse width trigger
	RUNT	Runt trigger
	RISEFall	Rise and Fall trigger
	FFTPeak	FFT Peak trigger
	LOGic	Logic trigger
	BUS	Bus trigger
Example	:SEARCH:TRIGger:TYPe EDGe Sets the search trigger to the edge type.	

**:SEARCH:TRIGger:SOURce** 
 →  
 →

Description	Sets or queries the search trigger source.	
Syntax	:SEARCH:TRIGger:SOURce {CH1   CH2   CH3   CH4   D0   D1   D2   D3   D4   D5   D6   D7   D8   D9   D10   D11   D12   D13   D14   D15   ? }	
Parameter/Return parameter	CH1 to CH4	Channel 1 to Channel 4
	D0 to D15	Digital channels D0 to D15
Example	:SEARCH:TRIGger:SOURce CH1 Sets the search trigger source as CH1.	

**:SEARCH:TRIGger:EDGE:SLOP** 
 →  
 →

Description	Sets or queries the search trigger slope.	
-------------	---	--

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

Related commands :SEARCH:TRIGger:TYPE

Parameter	RISe	Rising slope
	FALL	Falling slope
	EITHER	Either rising or falling slope

Return parameter Returns the trigger slope.

Example :SEARCH:TRIGger:EDGE:SLOP FALL  
Sets the search trigger slope to falling.

Set →

→ Query

### :SEARCH:TRIGger:LEVEL

Description Sets or queries the search trigger level.

Syntax :SEARCH:TRIGger:LEVEL {TTL | ECL | SETTO50 | <NRf> | ?}

Related commands :SEARCH:TRIGger:TYPE

Parameter	<NRf>	Trigger level value
	TTL	Sets the search trigger level to TTL.
	ECL	Sets the search trigger level to ECL.
	SETTO50	Sets the search trigger level to the User level (50% by default).


Return parameter <NR3> Returns the trigger.

Example1 :SEARCH:TRIGger:LEVEL TTL  
Sets the search trigger level to TTL.

Example2 :SEARCH:TRIGger:LEVEL 3.30E-1  
Sets the search trigger level to 330mV/mA.

**:SEARCH:TRIGger:HLEVel** 


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

**:SEARCH:TRIGger:LLEVel** 


Description	Sets or queries the low level search trigger.	
Note	Applicable for Rise and Fall/Pulse Runt triggers.	
Syntax	:SEARCH:TRIGger:LLEVel { <NRf>   ?}	
Related commands	:SEARCH:TRIGger:TYPE	
Parameter	<NRf>	Low level value.
Return parameter	<NR3>	Returns the low level.
Example	:SEARCH:TRIGger:LLEVel -3.30E-3 Sets the low level search trigger to 330mV/mA.	

**:SEARCH:TRIGger:PULSEWidth:POLarity** 


Description	Sets or queries the pulse width search trigger polarity.	
Syntax	:SEARCH:TRIGger:PULSEWidth:POLarity { POSitive   NEGative   ?}	

Related commands :SEARCH:TRIGger:TYPE

Parameter	POSitive	Positive polarity
	NEGative	Negative polarity

Return parameter Returns the pulse width polarity.

Example :SEARCH:TRIGger:PULSEWidth:POLarity POSitive  
Sets the pulse width polarity to positive.

Set →

:SEARCH:TRIGger:RUNT:POLarity → Query

Description Sets or queries the Pulse Runt search trigger polarity.

Syntax :SEARCH:TRIGger:RUNT:POLarity {POSitive | NEGative | EITHer | ?}

Related commands :SEARCH:TRIGger:TYPE

Parameter	POSitive	Positive polarity
	NEGative	Negative polarity
	EITHer	Positive or negative polarity

Return parameter Returns the pulse runt search trigger polarity.

Example :SEARCH:TRIGger:RUNT:POLarity POSitive  
Sets the Pulse Runt search trigger polarity to positive.

Set →

:SEARCH:TRIGger:RISEFall:SLOP → Query

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

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

Related commands :SEARCH:TRIGger:TYPE

Parameter	RISe	Rising slope
	FALL	Falling slope
	EITher	Either rising or falling slope

Return parameter Returns the rise & fall slope.

Example :SEARCH:TRIGger:RISEFall :SLOP RISe  
Sets the Rise & Fall search trigger slope to rising.

Set →

→ Query

**:SEARCH:TRIGger:PULSe:WHEn**

Description Sets or queries the pulse width search trigger conditions.

Syntax :SEARCH:TRIGger:PULSe:WHEn {MOREthan | LESSthan | EQual | UNEQual | ?}

Related commands :SEARCH:TRIGger:TYPE  
:SEARCH:TRIGger:PULSe:TIME

Parameter	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 →

:SEARCH:TRIGger:RUNT:WHEn

→ Query

Description Sets or queries the pulse runt search trigger conditions.

Syntax :SEARCH:TRIGger:RUNT:WHEn {MOREthan | LESSthan | EQUAL | UNEQUAL | ? }

Related commands :SEARCH:TRIGger:TYPE  
:SEARCH:TRIGger:RUNT:TiMe

Parameter	MOREthan	>
	LESSthan	<
	Equal	=
	UNEQUAL	≠

Return parameter Returns the pulse runt search trigger conditions.

Example :SEARCH:TRIGger:RUNT:WHEn UNEQUAL  
Sets the pulse runt search trigger condition to unequal (≠).

Set →

:SEARCH:TRIGger:RUNT:TiMe

→ Query

Description Sets or queries the pulse runt search trigger time.

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

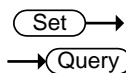
Related commands :SEARCH:TRIGger:TYPE  
:SEARCH:TRIGger:RUNT:WHEn

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 |? }

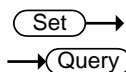
Related commands :SEARCH:TRIGger:TYPE  
:SEARCH:TRIGger:RISEFall:TIME

Parameter	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



Description Sets or queries the rise and fall time.

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

Related commands :SEARCH:TRIGger:TYPE  
:SEARCH:TRIGger:RISEFall:WHEn

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

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

**:SEARCH:TRIGger:BUS:TYPE**

→ **Query**

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

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

**Set** →

→ **Query**

Description	Sets or queries the I <sup>2</sup> 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 <sup>2</sup> C search trigger condition.
	STOP	Set Stop as the I <sup>2</sup> C search trigger condition.
	REPEATstart	Set Repeat of Start as the I <sup>2</sup> C search trigger condition.
	ACKMISS	Set Missing Acknowledgement as the I <sup>2</sup> C search trigger condition.
	ADDRess	Set Address as the I <sup>2</sup> C search trigger condition.



DATA	Set Data as the I <sup>2</sup> C search trigger condition.
ADDRANDDATA	Set Address and Data as the I <sup>2</sup> C search trigger condition.

Return parameter Returns the I<sup>2</sup>C bus search trigger condition.

Example :SEARCH:TRIGger:BUS:B1:I2C:CONDition ADDRess  
Set Address as the I<sup>2</sup>C search trigger condition.

(Set) →

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

Description Sets or queries the I<sup>2</sup>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.

(Set) →

:SEARCH:TRIGger:BUS:B1:I2C:ADDRess:TYPE → (Query)

Description Sets the I<sup>2</sup>C bus address type, or queries what the setting is for the search trigger.

Syntax :SEARCH:TRIGger:BUS:B1:I2C:ADDRess:TYPE {GENeralcall | STARtbyte | HSmode | EEPROM | CBUS | ? }

Related commands :SEARCH:TRIGger:BUS:B1:I2C:CONDition

Parameter	GENeralcall	Set a general call address (0000 000 0).
	STARtbyte	Set a start byte address. (0000 000 1)
	HSmode	Set a high-speed mode address. (0000 1xx x)
	EEPROM	Set an EEPROM address. (1010 xxx x)
	CBUS	Set a CBUS address. (0000 001 x)

Return Parameter Returns the address type

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

Set →

:SEARCH:TRIGger:BUS:B1:I2C:ADDRes:VALue → Query

Description Sets or queries the I<sup>2</sup>C bus address value when the I<sup>2</sup>C 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
-----------	---------	--

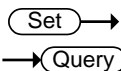
Return Parameter Returns the address value in binary.

```
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
:DIRrection
```



**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:DIRrection { READ | WRITE | NOCARE | ? }

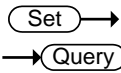
**Related commands** :SEARCH:TRIGger:BUS:B1:I2C:CONDition

<b>Parameter</b>	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:DIRrection
              READ
              Sets the direction to READ.
```

```
:SEARCH:TRIGger:BUS:B1:I2C:DATa:SIZE
```



**Description** Sets or queries the data size in bytes for the I<sup>2</sup>C bus.

**Note** This setting only applies when the I<sup>2</sup>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<sup>2</sup>C bus when the I<sup>2</sup>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   TXPARltyerr   RXPARltyerr   ? }	
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.
	RXPARltyerr	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.
	TXPARltyerr	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.

Set →

:SEARCH:TRIGger:BUS:B1:UART:RX:DATA:VALue → 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.


Example1 :SEARCH:TRIGger:BUS:B1:UART:CONDition RXDATA  
:SEARCH:TRIGger:BUS:B1:UART:RX:DATA:SIZE 1  
:SEARCH:TRIGger:BUS:B1:UART:RX:DATA:VALue "1x1x0101"

Sets the value to 1x1x0101

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

(Set) →

:SEARCH:TRIGger:BUS:B1:UART:TX:DATA:SIZE → (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.	

(Set) →

:SEARCH:TRIGger:BUS:B1:UART:TX:DATA: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	<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 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

:SEARCH:TRIGger:BUS:B1:SPI:CONDition           
Set →  
 → Query

Description       Sets or queries the SPI search triggering condition.

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


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

Return Parameter Returns the triggering condition.

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

:SEARCH:TRIGger:BUS:B1:SPI:DATA:SIZE           
Set →  
 → 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.	

Set →  
 :SEARCH:TRIGger:BUS:B1:SPI:DATA:MISO: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	

Set →  
 → Query

**: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

<b>Parameter</b>	<code>&lt;sting&gt;</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

Set →  
 → Query

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

**Description**      Sets or returns the CAN search trigger condition.

**Syntax**            :SEARCH:TRIGger:BUS:B1:CAN:CONDition {SOFF|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

:SEARCH:TRIGger:BUS:B1:CAN:FRAMeType 

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.

Set →

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

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

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


Parameter/	STANDard	Standard addressing mode
Return parameter	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

Set →

:SEARCH:TRIGger:BUS:B1:CAN:IDentifier:VALue → Query

Description Sets or returns the identifier string used for the CAN search trigger.

 Note Only applicable when the search trigger condition is set to ID or IDANDDATA.


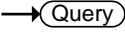
Syntax :SEARCH:TRIGger:BUS:B1:CAN:IDentifier:VALue {<string>|?}

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

Parameter/	<string>	The size of the string depends on the data size setting. The string must be enclosed in double quotes, "string".
Return parameter		

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

Set →

→ Query

**: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|?}

<b>Parameter/ Return parameter</b>	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** (Set) →  
→ (Query)

Description	Sets or returns the length of the data string in bytes for the CAN search trigger.	
 Note	Only applicable when the condition is set to DATA or IDANDDATA.	
Syntax	:SEARCH:TRIGger:BUS:B1:CAN:DATA:SIZE {<NR1>[?]}	
Parameter/ Return parameter	<NR1>	1~8 (bytes)
Example	:SEARCH:TRIGger:BUS:B1:CAN:DATA:SIZE?>1 :SEARCH:TRIGger:BUS:B1:CAN:DATA:SIZE 2 :SEARCH:TRIGger:BUS:B1:CAN:DATA:SIZE?>2	

**:SEARCH:TRIGger:BUS:B1:CAN:DATA:VALue** (Set) →  
→ (Query)

Description	Sets or returns the binary data string to be used for the CAN search trigger.	
Related Commands	:SEARCH:TRIGger:BUS:B1:CAN:DATA:SIZE	
Syntax	:SEARCH:TRIGger:BUS:B1:CAN:DATA:VALue {<string>[?]}	
Parameter/ Return parameter	<string>	The size of the string depends on the data size setting. The string must be enclosed in double quotes, "string". String contents: x = don't care 1 = binary 1 0 = binary 0



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

```
: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 S LEEP 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
```

Set →

:SEARCH:TRIGger:BUS:B1:LIN:DATa:QUALifier → Query

Description	Sets or returns the LIN data qualifier.	
 Note	Only applicable when the search trigger condition is set to DATA or IDANDDATA.	
Syntax	:SEARCH:TRIGger:BUS:B1:LIN:DATa:QUALifier {LESSthan MOREthan EQUAL UNEQUAL LESSEQUAL MOREEQUAL ?}	
Parameter/ Return parameter	LESSthan	Sets search to trigger when the data is less than the qualifier value.
	MOREthan	Sets search to trigger when the data is greater than the qualifier value.
	EQUAL	Sets search to trigger when the data is equal to the qualifier value.
	UNEQUAL	Sets search to trigger when the data is not equal to the qualifier value.
	LESSEQUAL	Sets search to trigger when the data is less than or equal to the qualifier value.
	MOREEQUAL	Sets search to trigger when the data is more than or equal to the qualifier value.
Example	:SEARCH:TRIGger:BUS:B1:LIN:DATa:QUALifier?>EQUAL :SEARCH:TRIGger:BUS:B1:LIN:DATa:QUALifier MOREthan :SEARCH:TRIGger:BUS:B1:LIN:DATa:QUALifier?>MORETHAN	

**:SEARCH:TRIGger:BUS:B1:LIN:DATA:SIZE** 


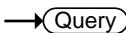


**Description** Sets or returns the length of the data string in bytes for the LIN search trigger.  
 Note: Only applicable when the condition is set to DATA or IDANDDATA.


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

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

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

**:SEARCH:TRIGger:BUS:B1:LIN:DATA:VALue** 



**Description** Sets or returns the binary data string to be used for the LIN search trigger.

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

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

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

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

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

Set →

```
: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|?}

<b>Parameter/ Return parameter</b>	SYNC	Sets the LIN error type to SYNC.
	PARity	Sets the LIN error type to parity.
	CHECKsum	Sets the LIN error type to checksum.

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

Set →

```
:SEARCH:TRIGger:BUS:B1:LIN:IDentifier:VALue → Query
```

**Description** Sets or returns the identifier string to be used for the LIN search trigger.



**Note**

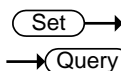
Only applicable when the condition is set to ID or IDANDDATA.

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

<b>Parameter/ Return parameter</b>	<string>	The size of the string depends on the data size setting. The string must be enclosed in double quotes, "string".
--	----------	--

String contents:  
 x = don't care  
 1 = binary 1  
 0 = binary 0

Example :SEARCH:TRIGger:BUS:B1:LIN:CONDition ID  
 :SEARCH:TRIGger:BUS:B1:LIN:IDentifier:VALue  
 "00X1X01X"  
 :SEARCH:TRIGger:BUS:B1:LIN:IDentifier:VALue?  
 >01100X1X01X



**:SEARCH:FFTPeak:METHOD**

Description Sets or returns the FFT peak method type.

Related :SEARCH:TRIGger:TYPE  
 Commands :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 →  
 → Query

**:SEARCH:FFTPeak:METHOD:MPEak**

---

Description	Sets the active peak number (1 ~ 10) or return the frequency of the active peak number.	
Related Commands	:SEARCH:TRIGger:TYPE :SEARCH:FFTPeak:METHOD	
Syntax	:SEARCH:FFTPeak:METHOD:MPEak {<NR1>   ?}	
Parameter	<NR1>	Active peak number.
Return parameter	<NR3>	Frequency of the active peak.
Example	:SEARCH:FFTPeak:METHOD MPEak :SEARCH:FFTPeak:METHOD? >MPEAK :SEARCH:FFTPeak:METHOD:MPEak? >1.000E+00 :SEARCH:FFTPeak:METHOD:MPEak 2 :SEARCH:FFTPeak:METHOD:MPEak? >2.000E+00	

Set →  
 → Query

**: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	:SEARCH:FFTPeak:SINFo? >PEAK :SEARCH:FFTPeak:SINFo mark :SEARCH:FFTPeak:SINFo? >MARK	

**:SEARCH:FFTPeak:LIST**

→ Query

Description	Returns the data of the search event table.
Syntax	:SEARCH:FFTPeak:LIST?
Example	:SEARCH:FFTPeak:LIST? No.,Frequency,Value; 1,1.000E+04,-6.400E+00; 2,2.750E+06,-7.360E+01; 3,2.830E+06,-7.280E+01; 4,2.910E+06,-7.200E+01; 5,3.020E+06,-7.120E+01; 6,3.170E+06,-7.040E+01; 7,5.550E+06,-8.240E+01; 8,5.640E+06,-8.160E+01; 9,5.740E+06,-8.080E+01; 10,5.900E+06,-8.000E+01;

Set →

**:SEARCH:FFTPeak:SOURce**

→ Query

Description	Sets or returns the fft peak source.
Related Commands	:SEARCH:FFTPeak:MEthod :SEARCH:FFTPeak:MEthod:MPeak :SEARCH:FFTPeak:SIInfo :SEARCH:FFTPeak:LIST When in SA Mode, the SA source affects the operation object of the above command.
Syntax	:SEARCH:FFTPeak:SOURce {SA1 SA2} :SEARCH:FFTPeak:SOURce?
Parameter	SA1 Sets the fft peak source as SA1. SA2 Sets the fft peak source as SA2.

Example :SEARCH:FFTPeak: SOURce SA2  
 :SEARCH:FFTPeak: SOURce?  
 SA2

Set →

:SEARCH:TRIGger:LOGic:INPut:CLOCK:SOURce → Query

Description Sets or returns the channel to use as the clock source.



Note

When the clock source selects NONE, it will become pattern trigger.

Syntax :SEARCH:TRIGger:LOGic:INPut:CLOCK:SOURce  
 {NONE|D0|D1|D2|D3|D4|D5|D6|D7|D8|D9|D10|D11|  
 D12|D13|D14|D15}  
 :SEARCH:TRIGger:LOGic:INPut:CLOCK:SOURce?

Parameter	NONE	Set a Pattern trigger
	D0~D15	Set the digital input channel source.

Set →

:SEARCH:TRIGger:LOGic:INPut:CLOCK:EDGE → Query

Description Sets the polarity of the clock channel.


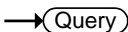
Related Commands :SEARCH:TRIGger:LOGic:INPut:CLOCK:SOURce


Syntax :SEARCH:TRIGger:LOGic:INPut:CLOCK:EDGE  
 {RISe|FALL|EITher}  
 :SEARCH:TRIGger:LOGic:INPut:CLOCK:EDGE?

Parameter	RISe	Rising slope
	FALL	Falling slope
	EITher	Either slope

Example :SEARCH:TRIGger:LOGic:INPut:CLOCK:SOURce D0  
 :SEARCH:TRIGger:LOGic:INPut:CLOCK:EDGE FALL



:SEARCH:TRIGger:LOGic:FUNcTion		 
Description	Sets or queries the logical combination of the input channels for logic trigger search.	
Related Commands	:SEARCH:TRIGger:LOGic:PATtern:INPut:D<X>	
Syntax	:SEARCH:TRIGger:LOGic:FUNcTion {AND NAND NOR OR} SEARCH:TRIGger:LOGic:FUNcTion?	
Parameter	AND	Sets the AND mode of define logic.
	NAND	Sets the NAND mode of define logic.
	NOR	Sets the NOR mode of define logic.
	OR	Sets the OR mode of define logic.
Example	SEARCH:TRIGger:LOGic:FUNcTion? AND SEARCH:TRIGger:LOGic:FUNcTion NAND Sets the NAND mode of define logic.	

:SEARCH:TRIGger:LOGic:PATtern		
Description	Returns the conditions used for generating an A logic pattern trigger search, with respect to the defined input pattern, and identifies the time that the selected pattern may be true and still generate the trigger search.	
Syntax	:SEARCH:TRIGger:LOGic:PATtern?	
Example	:SEARCH:TRIGGER:LOGIC:PATTERN:INPUT:D0 X;D1 X;D2 X;D3 X;D4 X;D5 X;D6 X;D7 X;D8 X;D9 X;D10 X;D11 X;D12 X;D13 X;D14 X;D15 X;;SEARCH :TRIGGER:LOGIC:PATTERN:WHEN TRUE;; SEARCH: TRIGGER:LOGIC:PATTERN:DELTATIME 1.000e-08;	

Set →  
 → Query

**:SEARCH:TRIGger:LOGic:PATtern:INPut:D<x>**

Description	Sets or returns the logic trigger search input for the specified digital channel.	
Syntax	:SEARCH:TRIGger:LOGic:PATtern:INPut:D<x> {HIGH LOW X} :SEARCH:TRIGger:LOGic:PATtern:INPut:D<x>?	
Parameter	<x>	It indicates the channel number.
	HIGH	Set the logic high state.
	LOW	Set the logic low state.
	X	Set a “don’t care” state.
Example	:SEARCH:TRIGger:LOGic:PATtern:INPut:D0? X	

Set →  
 → Query

**:SEARCH:TRIGger:LOGic:PATtern:DELTatime**

Description	Sets or returns the pattern trigger search delta time value.	
Related Commands	:SEARCH:TRIGger:LOGic:PATtern:WHEn	
Syntax	:SEARCH:TRIGger:LOGic:PATtern:DELTatime <NR3> :SEARCH:TRIGger:LOGic:PATtern:DELTatime?	
Parameter	<NR3>	It indicates a floating point value with exponent that sets the pattern trigger search time value. A range of 1E-9 (1 ns) to 10.0E0 (10s).
Example	:SEARCH:TRIGger:LOGic:PATtern:DELTatime 8.960e-05 :SEARCH:TRIGger:LOGic:PATtern:DELTatime? 8.960e-05	

**:SEARCH:TRIGger:LOGic:PATtern:WHEn** 



Description	Sets or returns the pattern logic condition on which to trigger search the oscilloscope.	
Related Commands	:SEARCH:TRIGger:LOGic:PATtern:DELtetime	
Syntax	:SEARCH:TRIGger:LOGic:PATtern:WHEn {TRUE FALSE LESSthan MORethan EQual UNEQUAL} :SEARCH:TRIGger:LOGic:PATtern:WHEn?	
Parameter	TRUE	Set true mode.
	FALSE	Set false mode.
	LESSTHAN	Set less than mode(Is True < time period(set in :SEARCH:TRIGger:LOGic:PATtern:DELtetime)).
	MORETHAN	Set more than mode(Is True > time period(set in :SEARCH:TRIGger:LOGic:PATtern:DELtetime)).
	EQUAL	Set equal mode(Is True = time period(set in :SEARCH:TRIGger:LOGic:PATtern:DELtetime)).
	UNEQUAL	Set unequal mode(Is True ≠ time period(set in :SEARCH:TRIGger:LOGic:PATtern:DELtetime)).
Example	:SEARCH:TRIGger:LOGic:PATtern:WHEn FALSE :SEARCH:TRIGger:LOGic:PATtern:WHEn? FALSE	

**:SEARCH:TRIGger:BUS:TYPE** 

Description	Returns the current bus trigger search type as I <sup>2</sup> C, SPI, UART or PARALLEL .
Syntax	:SEARCH:TRIGger:BUS:TYPE?

---

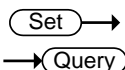
Parameter	I2C	I <sup>2</sup> C mode
	SPI	SPI mode
	UART	UART mode
	PARALLEL	Parallel mode

---

Example :SEARCH:TRIGger:BUS:TYPe?  
UART

## Label Commands

:CHANnel<X>:LABel.....	277
:CHANnel<X>:LABel:DISPlay .....	278
:REF<X>:LABel .....	278
:REF<X>:LABel:DISPlay .....	279
:BUS1:LABel.....	280
:BUS1:LABel:DISPlay .....	280
:D<x>:LABel .....	281
:D<x>:LABel:DISPlay.....	282
:DIGital:ANAlog:A<x>:LABel .....	282
:DIGital:ANAlog:A<x>:LABel:DISPlay .....	283
:SET<X>:LABel .....	284

: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 Set →  
→ Query

Description Turns the label on/off for the selected channel or returns its status.

Syntax :CHANnel<X>:LABel:DISPlay { OFF | ON | ? }

Related commands :CHANnel<X>:LABel

Parameter	<X>	Channel 1, 2, 3, 4
	OFF	Turns the file label off for the selected channel.
	ON	Turns the file label on for the selected channel.

Return parameter Returns the status of the file label for the selected channel (ON, OFF).

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

:REF<X>:LABel Set →  
→ Query

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



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 →

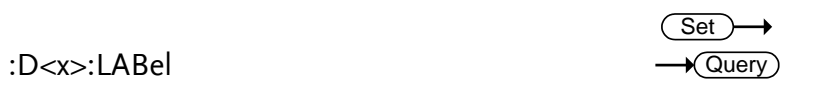
→ Query

**: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 :D<x>:LABel :D<x>:LABel:DISPlay :DIGital:ANALog:A<x>:LABel :DIGital:ANALog:A<x>:LABel:DISPlay
Parameter	OFF Turns the file label off for the bus. ON Turns the file label on for the bus.
Return parameter	Returns the status of the file label for the bus (ON, OFF).
Example	:BUS1:LABel "Bus" :BUS1:LABel:DISPlay ON :BUS1:LABel:DISPlay? ON Sets the label for the bus to "Bus" and then turns the label display on. The query return shows that the label is on.



Description	Sets or returns the waveform label for digital channels.
Syntax	:D<x>:LABel {<string>   ?}
Related commands	:D<x>:LABel:DISPlay
Parameter	<x> Digital channel number D0~D15 <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	<code>&lt;string&gt;</code>	Returns the label for the bus. No return indicates that there has not been a file label assigned for bus.
------------------	-----------------------------	---

Example           :D1:LABel "D1"  
                   Sets the digital channel 1 label as "D1".

Set →  
 → Query

**:D<x>:LABel:DISPlay**

Description	Turns the label on/off for the selected digital channel or returns its status.
-------------	--

Syntax	:D<x>:LABel:DISPlay { OFF   ON   ? }
--------	--------------------------------------

Related commands	:D<x>:LABel
------------------	-------------

Parameter	OFF	Turns the file label off for the selected digital channel.
	ON	Turns the file label on for the selected digital channel.

Return parameter	Returns the status of the file label for the digital channel (ON, OFF).
------------------	---

Example           :~:LABel "D1"  
                   :~:LABel?  
                   >D1  
                   :~:LABel:DISPlay ON  
                   D1:LABel:DISPlay?  
                   ON  
                   Sets the label for the D1 channel to "D1" and then turns the label display on. The query return shows that the label is on.

Set →  
 → Query

**:DIGital:ANALog:A<x>:LABel**

Description	Sets or returns the label for analog waveforms.
-------------	---

Syntax	:DIGital:ANALog:A<x>:LABel {<string>   ?}
--------	---

Related commands :DIGital:ANALog:A<x>:LABel:DISPlay

Parameter	<x>	Analog waveform number (1~2).
	<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.

Example :DIG:ANA:A1:LAB "A1"  
Sets the analog waveform 1 label as "A1".



Description Turns the label on/off for the analog waveform or returns its status.

Syntax :DIGital:ANALog:A<x>:LABel:DISPlay {OFF | ON | ? }

Related commands :DIGital:ANALog:A<x>:LABel

Parameter	OFF	Turns the file label off for the analog waveform.
	ON	Turns the file label on for the analog waveform.

Return parameter Returns the status of the file label for the analog waveform (ON, OFF).

Example :DIGital:ANALog:A1:LABel "A1"  
:DIGital:ANALog:A1:LABel:DISPlay ON  
:DIGital:ANALog:A1:LABel:DISPlay?  
ON  
Sets the label for the analog waveform to "A1"

and then turns the label display on. The query return shows that the label is on.

Set →

→ Query

: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.
------------------	----------	---

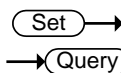
Example1             :SET1:LABel "SET1\_lab"  
 Sets the label for setup 1 as "SET1\_lab".

Example2             :SET1:LABel?  
 SET1\_lab

## Segment Commands

:SEGMENTS:STATE .....	285
:SEGMENTS:CURRENT .....	286
:SEGMENTS:TOTALNUM .....	286
:SEGMENTS:TIME .....	287
:SEGMENTS:DISPALL .....	287
:SEGMENTS:MEASURE:MODE .....	288
:SEGMENTS:MEASURE:PLOT:SOURCE .....	288
:SEGMENTS:MEASURE:PLOT:DIVIDE .....	289
:SEGMENTS:MEASURE:PLOT:SELECT .....	289
:SEGMENTS:MEASURE:PLOT:RESULTS.....	289
:SEGMENTS:MEASURE:TABLE:SOURCE .....	290
:SEGMENTS:MEASURE:TABLE:SELECT.....	291
:SEGMENTS:MEASURE:TABLE:LIST.....	291
:SEGMENTS:MEASURE:TABLE:SAVE .....	292
:SEGMENTS:SAVE .....	292
:SEGMENTS:SAVE:SOURCE .....	292
:SEGMENTS:SAVE:SELECT:START .....	293
:SEGMENTS:SAVE:SELECT:END .....	293
:SEGMENTS:MASK:STATE .....	294
:SEGMENTS:MASK:SOURCE .....	294
:SEGMENTS:MASK:CREATE .....	294
:SEGMENTS:MASK:UNITS .....	295

### :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.

Set →

### :SEGMents:DISPALL

→ 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.

Set →

→ Query

**:SEGMents:MEASure:MODE**

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.



**: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** 


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** 


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  

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 block data.
Syntax	:SEGMents:MEASure:TABLE:LIST?
Return parameter	Returns the measurements results as a block data for each segment.
Example	:SEGMents:MEASure:TABLE:LIST? >"GW-INSTEK, GDS-3654A, serial number PXXXXXX, version V1.37", Segment Summary : CH1, Seg., Pk-Pk (V), Pk-Pk (V), 1, 8.00m, 8.00m, 2, 8.00m, 8.00m, 3, 8.00m, 8.00m, 4, 8.00m, 8.00m, 5, 8.00m, 8.00m, 6, 8.00m, 8.00m, 7, 8.00m, 8.00m, 8, 8.00m, 8.00m, 9, 12.0m, 12.0m, 10, 8.00m, 8.00m,

**: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

Set →

**:SEGMents:SAVE:SOURce**

→ Query

**Description** Sets or queries the source segment waveform to save.

**Syntax** :SEGMents:SAVE:SOURce {CH1 | CH2 | CH3 | CH4 | D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | D8 | D9 | D10 | D11 | D12 | D13 | D14 | D15 | ? }

<b>Parameter/ Return parameter</b>	CH1~CH4	Channel 1 to 4.
	D0~D15	Digital channels D0~D15

**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.

Set →

:SEGMents:MASK:STATe

→ Query

Description Sets or queries the state of segment mask.



Note

It could not set on when the mask hasn't create.

Syntax :SEGMents:MASK:STATe {ON | OFF}  
 :SEGMents:MASK:STATe?

Parameter ON Turn on the segment mask.  
 OFF Turn off the segment mask.

Example :SEGMents:MASK:STATe ON  
 :SEGMents:MASK:STATe?  
 ON

Set →

:SEGMents:MASK:SOURce

→ Query

Description Sets or queries the referance source of segment mask.

Syntax :SEGMents:MASK:SOURce {CH1|CH2|CH3|CH4}  
 :SEGMents:MASK:SOURce?

Parameter CH1~CH4 Channel 1 to Channel 4

Example SEGMents:MASK:SOURce CH1  
 :SEGMents:MASK:SOURce?  
 CH1

Set →

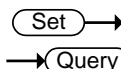
:SEGMents:MASK:CREATE

→ Query

Description Creates a segment mask fastly in according to the reference source or returns the setting of the mask.

Syntax	:SEGMents:MASK:CREATE {<Xmask>,<Ymask>} :SEGMents:MASK:CREATE?
Parameter	<Xmask> Sets the horizontal range for the mask. <Ymask> Sets the vertical range for the mask.
Example	:SEGMents:MASK:CREATE 0.1,0.2 Sets the range of mask as {0.1,0.2} and creates a mask.

**:SEGMents:MASK:UNITs**



Description	Sets or queries the units of segment mask.
Syntax	:SEGMents:MASK:UNITs {DIVIsions CURRent} :SEGMents:MASK:UNITs?
Parameter	DIVIsions Divisions units CURRent Current units
Example	:SEGMents:MASK:UNITs CURRent :SEGMents:MASK:UNITs? CURRent

## DVM Commands

The DVM commands are only available when the optional DVM software is installed.

:DVM:STATE .....	296
:DVM:SOURce .....	296
:DVM:MODE .....	297
:DVM:VALue .....	297

### :DVM:STATE

Set →

→ Query

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.	


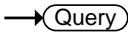
Set →


→ Query

### :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.	



 											
<b>:DVM:MODE</b>											
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.										

	
<b>:DVM:VALue</b>	
Description	Returns the measurement value of the selected mode.
Syntax	:DVM:VALue?
Related commands	:DVM:SOURce :DVM:STATE :DVM: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.....	298
:GONogo:EXECute .....	298
:GONogo:FUNCTion.....	299
:GONogo:NGCount .....	299
:GONogo:NGDefine.....	299
:GONogo:SOURce .....	300
:GONogo:VIOLation .....	300
:GONogo:SCRipt .....	300
:TEMPlate:MODE .....	301
:TEMPlate:MAXimum.....	301
:TEMPlate:MINimum.....	301
:TEMPlate:POSition:MAXimum.....	302
:TEMPlate:POSition:MINimum .....	302
:TEMPlate:SAVe:MAXimum .....	302
:TEMPlate:SAVe:MINimum .....	303
:TEMPlate:TOLerance.....	303
:TEMPlate:SAVe:AUTo .....	303

### :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.

(Set) →

**:GONogo:NGDefine** → (Query)

Description Sets the Go/NoGo “When” conditions.

Syntax :GONogo:NGDefine {EXITs|ENTers}[?]

Parameter/ Return Parameter	EXITs	Sets the NoGo condition to when the input signal exceeds the limit boundary.
	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

<b>:GONogo:SOURce</b>	
<b>Description</b>	Sets the source for the Go/NoGo signal.
<b>Syntax</b>	:GONogo:SOURce {CH1 CH2 CH3 CH4 ?}
<b>Parameter/ Return Parameter</b>	CH1~CH4
<b>Example</b>	:GONogo:SOURce CH1 Sets the source to CH1.

Set →  
 → Query

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

Set →

<b>:GONogo:SCRipt</b>	
<b>Description</b>	Activates/Deactivates the Go/NoGo APP or queries its state.
<b>Syntax</b>	:GONogo:SCRipt {OFF   ON   ?}
<b>Parameter/ Return Parameter</b>	ON Turns Go/NoGo APP on. OFF Turns the Go/NoGo APP off.
<b>Example</b>	: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.

Set →

:TEMPlate:POSition:MAXimum

→ 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.

Set →

:TEMPlate:POSition:MINimum

→ Query

Description Sets or queries the position of the minimum template.

Syntax :TEMPlate:POSition:MINimum {<NR2>|?}

Parameter <NR2> Desired template position (-12.0 ~ +12.0 divisions)

Return parameter Returns the position in the following format: "<NR2>Div"

Example :TEMPlate:POSition:MINimum 3.00  
Sets the minimum template position to 3.00 divisions.

:TEMPlate:SAVE:MAXimum

Set →

Description Saves the maximum template.

Syntax :TEMPlate:SAVE:MAXimum

**:TEMPlate:SAVe:MINimum****Set** →

Description	Saves the maximum template.
-------------	-----------------------------

Syntax	:TEMPlate:SAVe:MINimum
--------	------------------------

**Set** →**:TEMPlate:TOLerance**→ **Query**

Description	Sets or queries the tolerance as a percentage.
-------------	--

Syntax	:TEMPlate:TOLerance {<NR2> ?}
--------	-------------------------------

Parameter/ Return Parameter	<NR2>	The auto tolerance range (0.4% ~ 40%)
--------------------------------	-------	---------------------------------------

Example	:TEMPlate:TOLerance 10 Sets the tolerance to 10%.
---------	--

**:TEMPlate:SAVe:AUTO****Set** →

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

Syntax	:TEMPlate:SAVe:AUTO
--------	---------------------

## AWG Commands

:AWG:UTIL.....	305
:AWG:UTIL:AMPcpl .....	305
:AWG:UTIL:FREQcpl.....	306
:AWG:UTIL:FREQcpl:OFFSet .....	306
:AWG:UTIL:FREQcpl:RATio.....	307
:AWG:UTIL:TRACking.....	307
:AWG<x>:AMPlitude.....	307
:AWG<x>:FREQuency.....	308
:AWG<x>:FUNCTion .....	308
:AWG<x>:OFFSet .....	309
:AWG<x>:OUTPut:LOAd:IMPEDance .....	309
:AWG<x>:OUTPut:STATE .....	310
:AWG<x>:PHAsE.....	310
:AWG<x>:PULSe:DUTYcycle .....	310
:AWG<x>:RAMP:SYMmetry .....	311
:AWG<x>:MODulation:STATE .....	311
:AWG<x>:MODulation:TYPe.....	311
:AWG<x>:MODulation:AM:DEPth .....	312
:AWG<x>:MODulation:AM:FREQ.....	312
:AWG<x>:MODulation:AM:SHApe.....	312
:AWG<x>:MODulation:AM:PHAsE .....	313
:AWG<x>:MODulation:AM:DUTYcycle .....	313
:AWG<x>:MODulation:AM:SYMmetry .....	313
:AWG<x>:MODulation:AM:RATE .....	314
:AWG<x>:MODulation:FM:DEV .....	314
:AWG<x>:MODulation:FM:FREQ.....	315
:AWG<x>:MODulation:FM:SHApe .....	315
:AWG<x>:MODulation:FM:PHAsE .....	316
:AWG<x>:MODulation:FM:DUTYcycle.....	316
:AWG<x>:MODulation:FM:SYMmetry .....	316
:AWG<x>:MODulation:FM:RATE.....	317
:AWG<x>:MODulation:FSK:FREQ .....	317

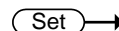


:AWG<x>:MODulation:FSK:RATE .....318  
 :AWG<x>:SWEep:STATE .....318  
 :AWG<x>:SWEep:TYPe .....318  
 :AWG<x>:SWEep:START .....319  
 :AWG<x>:SWEep:STOP .....319  
 :AWG<x>:SWEep:TIME.....319  
 :AWG<x>:SWEep:SPAN.....320  
 :AWG<x>:SWEep:CENTEr .....320  
 :AWG<x>:ARBitrary:EDIT:NUMPoint .....320  
 :AWG<x>:ARBitrary:EDIT:FUNCTion.....321  
 :AWG<x>:ARBitrary:SAVe:WAVEform .....321  
 :AWG<x>:ARBitrary:LOAd:WAVEform .....322  
 :AWG<x>:ARBitrary:EDIT:COpy.....322  
 :AWG<x>:ARBitrary:EDIT:CLEAr.....323  
 :AWG<x>:ARBitrary:EDIT:LINE .....323  
 :AWG<x>:ARBitrary:EDIT:SCALE.....324  
 :AWG<x>:ARBitrary:EDIT:POINt.....324  
 :AWG<x>:ARBitrary:EDIT:POINt:ADD.....324  
 :AWG<x>:ARBitrary:EDIT:POINt:DELEte.....325

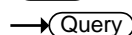
**:AWG:UTIL**



Description	Reset all of the AWG settings to the default.
Syntax	:AWG:UTIL{PRESet}
Parameter	PRESet Set the AWG settings to default.
Example	:AWG:UTIL PRESet



**:AWG:UTIL:AMPCpl**



Description	Set or return the state of amplitude couple.
Syntax	:AWG:UTIL:AMPCpl{ON OFF} :AWG:UTIL:AMPCpl?

Parameter	ON	Turn on the amplitude couple.
	OFF	Turn off the amplitude couple.
Example	:AWG:UTIL:AMPCpl ON :AWG:UTIL:AMPCpl? >ON	

Set →

→ Query

**:AWG:UTIL:FREQCpl**

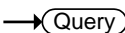
Description	Set or return the type of frequency couple.	
Syntax	:AWG:UTIL:FREQCpl {OFF  OFFSet  RATio} :AWG:UTIL:FREQCpl?	
Parameter	OFF	Turn off the Frequency Couple.
	OFFSet	The frequency of Gen. 1 and Gen. 2 are coupled by offset.
	RATio	The frequency of Gen. 1 and Gen. 2 are coupled by ratio.
Example	:AWG:UTIL:FREQCpl RATio :AWG:UTIL:FREQCpl? >RATIO	

Set →

→ Query

**:AWG:UTIL:FREQCpl:OFFSet**

Description	Set or return the frequency offset between Gen. 1 and Gen. 2 for frequency couple	
Syntax	:AWG:UTIL:FREQCpl:OFFSet{<NRf>} :AWG:UTIL:FREQCpl:OFFSet?	
Parameter	<NRf>	Value of offset.
Example	:AWG:UTIL:FREQCpl:OFFSet 50 :AWG:UTIL:FREQCpl:OFFSet? >5.00000e+01	

**:AWG:UTIL:FREQCpl:RATio**

Description	Set or return the frequency ratio between Gen. 1 and Gen. 2 for frequency couple.
Syntax	:AWG:UTIL:FREQCpl:RATio{<NRF>} :AWG:UTIL:FREQCpl:RATio?
Parameter	<NRF> Value of ratio.
Example	:AWG:UTIL:FREQCpl:RATio 2.5 :AWG:UTIL:FREQCpl:RATio? >2.50000e+00


**:AWG:UTIL:TRACking**

Description	Set or return the state of AWG tracking.
Syntax	:AWG:UTIL:TRACking{ON OFF} :AWG:UTIL:TRACking?
Parameter	ON Turn on the AWG tracking. OFF Turn off the AWG tracking.
Example	:AWG:UTIL:TRACking ON :AWG:UTIL:TRACking? >ON


**:AWG<x>:AMPlitude**

Description	Sets or returns the waveform amplitude.
Syntax	:AWG<x>:AMPlitude {<NRF>   ?}
Related command	:AWG<x>:OUTPut:LOAD:IMPEDance
Parameter/	<x> Channel number 1~2.

Return parameter	<Nrf>	Amplitude in Volts. (50Ω impedance 0.1~2.5V) (High Z impedance 0.2~5V)
------------------	-------	--

Example :AWG1:AMP 1  
:AWG1:AMPlitude?  
1.00000e+00

Set →

→ Query

**:AWG<x>:FREQuency**

Description	Sets or returns the waveform frequency.	
Syntax	:AWG<x>:FREQuency {<Nrf>   ?}	
Parameter/ Return parameter	<x> <Nrf>	Channel number 1~2. Frequency in Hertz.

Example :AWG1:FREQ 2000  
:AWG1:FREQuency?  
2.00000e+03

Set →

→ Query

**:AWG<x>:FUNCTion**

Description	Sets or returns the type of waveform.	
Syntax	:AWG<x>:FUNCTion {ARBitrary   SINE   SQUAre   PULSe   RAMP   DC   NOISe   SINC   GAUSSian   LORENTz   EXPRise   EXPFall   HAVERSINE   CARDIac   ?}	
Parameter/ Return parameter	<x> ARBitrary SINE SQUAre PULSe RAMP DC	Channel number 1~2. Arbitrary waveform Sine waveform Square waveform Pulse waveform Ramp waveform DC waveform

NOISe	Noise waveform
SINC	Sinc waveform
GAUSsian	Gaussian waveform
LORENTz	Lorentz waveform
EXPRise	Exponential rise waveform
EXPfall	Exponential fall waveform
HAVERSINe	Haversine waveform
CARDIac	Cardiac waveform

Example :AWG1:FUNC?  
>SINE

:AWG<x>:OFFSet (Set) →  
→ (Query)

Description Sets or returns the waveform offset.

Syntax :AWG<x>:OFFSet {<NRf> | ?}

Parameter/ Return parameter	<x>	Channel number 1~2.
	<NRf>	Offset in Volts.

Example :AWG1:OFFSet?  
0.00000e+00  
:AWG1:OFFSet 1  
:AWG1:OFFSet?  
1.00000e+00

:AWG<x>:OUTPut:LOAD:IMPEDance (Set) →  
→ (Query)

Description Sets or returns the output termination

Syntax :AWG<x>:OUTPut:LOAD:IMPEDance {FIFty | HIGHZ | ?}

Parameter/ Return parameter	<x>	Channel number 1~2
	FIFty	50 Ohm output termination

**HIGHZ** High Z output termination

**Example** :AWG1:OUTP:LOA:IMPED HIGHZ  
 Sets the output termination of channel 1 to high impedance.

Set →

:AWG<x>:OUTPut:STATE → Query

**Description** Sets or returns the channel output state.

**Syntax** :AWG<x>:OUTPut:STATE {OFF | ON | ?}

<b>Parameter/ Return parameter</b>	<x>	Channel number 1~2
	OFF	Turns the channel output off
	ON	Turns the channel output on

**Example** :AWG1:OUTP:STATE OFF  
 Turns the channel 1 output off.

Set →

:AWG<x>:PHAsE → Query

**Description** Sets or returns the channel phase.

**Syntax** :AWG<x>:PHAsE {<NRf> | ?}

<b>Parameter/ Return parameter</b>	<x>	Channel number 1~2.
	<NRf>	Phase in degree -180~180°

**Example** :AWG1:PHA 45  
 Sets the channel 1 phase to 45°.

Set →

:AWG<x>:PULSe:DUTYcycle → Query

**Description** Sets or returns the pulse duty cycle.

**Syntax** :AWG<x>:PULSe:DUTYcycle {<NRf> | ?}

<b>Parameter/ Return parameter</b>	<x>	Channel number 1~2.
	<NRf>	Duty cycle in percentage 0.2~99.8%

Example :AWG1:PULS:DUTY 50  
Sets the channel 1 pulse duty cycle to 50%.

:AWG<x>:RAMP:SYMmetry  

Description Sets or returns the ramp symmetry.

Syntax :AWG<x>:RAMP:SYMmetry {<NRf> | ?}

Parameter/ Return parameter	<x>	Channel number 1~2.
	<NRf>	Symmetry of the ramp waveform 0~100%

Example :AWG1:RAMP:SYM 15  
Sets the channel 1 ramp symmetry to 15%.


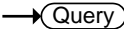
:AWG<x>:MODulation:STATE  

Description Sets or returns the modulation state.

Syntax :AWG<x>:MODulation:STATE {OFF | ON | ?}

Parameter/ Return parameter	<x>	Channel number 1~2.
	OFF	Sets the modulation to off.
	ON	Sets the modulation to on.

Example :AWG1:MOD:STATE ON  
Turns the modulation on for channel 1.

:AWG<x>:MODulation:TYPE  

Description Sets or returns the type of modulation.

Syntax :AWG<x>:MODulation:TYPE {AM | FM | FSK | ?}

Parameter/ Return parameter	<x>	Channel number 1~2.
	AM	Sets a AM modulation.
	FM	Sets a FM modulation.
	FSK	Sets a FSK modulation.

Example :AWG1:MOD:TYPE AM  
 Sets a AM modulation for channel 1.

:AWG<x>:MODulation:AM:DEPth Set →  
→ Query

Description	Sets or returns the AM modulation depth.	
Syntax	:AWG<x>:MODulation:AM:DEPth {<NRf>   ?}	
Parameter/ Return parameter	<x>	Channel number 1~2.
	<NRf>	AM depth in percentage 0~120%.

Example :AWG1:MOD:AM:DEP?  
 >1.20000e+02

:AWG<x>:MODulation:AM:FREQ Set →  
→ Query

Description	Sets or returns the AM modulation frequency.	
Syntax	:AWG<x>:MODulation:AM:FREQ {<NRf>   ?}	
Parameter/ Return parameter	<x>	Channel number 1~2.
	<NRf>	AM frequency in Hertz.

Example :AWG1:MOD:AM:FREQ 1000  
 Sets the AM frequency to 1kHz.

:AWG<x>:MODulation:AM:SHApe Set →  
→ Query

Description	Sets or returns the shape of the AM modulation.	
Syntax	:AWG<x>:MODulation:AM:SHApe {SINE   SQUare   PULSe   RAMP   NOISe   ?}	
Parameter/ Return parameter	<x>	Channel number 1~2.
	SINE	Sine wave shape.
	SQUare	Square wave shape.
	PULSe	Pulse wave shape.
	RAMP	Ramp wave shape.



**NOISe** Noise wave shape.

Example :AWG1:MOD:AM:SHA RAMP  
Sets a ramp shape to the AM modulating waveform.

:AWG<x>:MODulation:AM:PHase (Set) →  
→ (Query)

Description Sets or returns the phase of the AM modulation (sine wave shape only).

Syntax :AWG<x>:MODulation:AM:PHase {<NRf> | ?}

Parameter/ Return parameter <x> Channel number 1~2.  
<NRf> Phase in degree -180~180°.

Example :AWG1:MOD:AM:PHA?  
>-1.80000e+02

:AWG<x>:MODulation:AM:DUTYcycle (Set) →  
→ (Query)

Description Sets or returns the duty cycle of the AM modulation (pulse wave shape only).

Syntax :AWG<x>:MODulation:AM:DUTYcycle {<NRf> | ?}

Parameter/ Return parameter <x> Channel number 1~2.  
<NRf> Duty cycle in percentage 2~98%.

Example :AWG1:MOD:AM:DUTY 50  
Sets the duty cycle of the AM modulating waveform to 50%.

:AWG<x>:MODulation:AM:SYMMetry (Set) →  
→ (Query)

Description Sets or returns the symmetry of the AM modulation (ramp wave shape only).

Syntax :AWG<x>:MODulation:AM:SYMMetry {<NRf> | ?}

Parameter/	<x>	Channel number 1~2.
Return parameter	<NRf>	Symmetry in percentage 0~100%.

Example :AWG1:MOD:AM:SYM 50  
 Sets the symmetry of the AM modulating waveform to 50%.

Set →

:AWG<x>:MODulation:AM:RATE

→ Query

Description Sets or returns the rate of the AM modulation (noise wave shape only).

Syntax :AWG<x>:MODulation:AM:RATE {RATE10M | RATE5M | RATE1M | RATE500K | RATE100K | RATE50K | RATE10K | RATE5K | RATE1K | ?}

Parameter/	<x>	Channel number 1~2.
Return parameter	RATE10M	10MHz noise rate.
	RATE5M	5MHz noise rate.
	RATE1M	1MHz noise rate.
	RATE500K	500kHz noise rate.
	RATE100K	100kHz noise rate.
	RATE50K	50kHz noise rate.
	RATE10K	10kHz noise rate.
	RATE5K	5kHz noise rate.
	RATE1K	1kHz noise rate.

Example :AWG1:MOD:AM:RATE RATE5K  
 Sets the noise rate of the AM modulating waveform to 5kHz.

Set →

:AWG<x>:MODulation:FM:DEV

→ Query

Description Sets or returns the deviation of the FM modulation.

Syntax	:AWG<x>:MODulation:FM:DEV {<NRf>   ?}	
Parameter/ Return parameter	<x>	Channel number 1~2.
	<NRf>	Frequency deviation in Hertz.
Example	:AWG1:MOD:FM:DEV? >2.000000000e+02	

Set →

→ Query

**:AWG<x>:MODulation:FM:FREQ**

Description	Sets or returns the frequency of the FM modulation.	
Syntax	:AWG<x>:MODulation:FM:FREQ {<NRf>   ?}	
Parameter/ Return parameter	<x>	Channel number 1~2.
	<NRf>	Frequency in Hertz.
Example	:AWG1:MOD:FM:FREQ 1000  Sets the frequency of the FM modulating waveform to 1kHz.	

Set →

→ Query

**:AWG<x>:MODulation:FM:SHApe**

Description	Sets or returns the shape of the FM modulation.	
Syntax	:AWG<x>:MODulation:FM:SHApe {SINE   SQUare   PULSe   RAMP   NOISe   ?}	
Parameter/ Return parameter	<x>	Channel number 1~2.
	SINE	Sine wave shape.
	SQUare	Square wave shape.
	PULSe	Pulse wave shape.
	RAMP	Ramp wave shape.
	NOISe	Noise wave shape.
Example	:AWG1:MOD:FM:SHA SINE  Sets a sine shape to the FM modulation.	

Set →  
 → Query

**:AWG<x>:MODulation:FM:PHase**

Description	Sets or returns the phase of the FM modulation (sine wave shape only).	
Syntax	:AWG<x>:MODulation:FM:PHase {<Nrf>   ?}	
Parameter/ Return parameter	<x>	Channel number 1~2.
	<Nrf>	Phase in degree -180~180°.
Example	:AWG1:MOD:FM:PHA 90 Sets a 90° phase to the FM modulating waveform.	

Set →  
 → Query

**:AWG<x>:MODulation:FM:DUTYcycle**

Description	Sets or returns the duty cycle of the FM modulation (pulse shape wave only).	
Syntax	:AWG<x>:MODulation:FM:DUTYcycle {<Nrf>   ?}	
Parameter/ Return parameter	<x>	Channel number 1~2.
	<Nrf>	Duty cycle in percentage 1~99%.
Example	:AWG1:MOD:FM:DUTY 50 Sets the duty cycle of the FM modulating waveform to 50%.	

Set →  
 → Query

**:AWG<x>:MODulation:FM:SYMmetry**

Description	Sets or returns the symmetry of the FM modulation (ramp shape wave only).	
Syntax	:AWG<x>:MODulation:FM:SYMmetry {<Nrf>   ?}	
Parameter/ Return parameter	<x>	Channel number 1~2.
	<Nrf>	Symmetry in percentage 0~100%.
Example	:AWG1:MOD:FM:SYM 50 Sets the symmetry of the FM modulating waveform to 50%.	

**:AWG<x>:MODulation:FM:RATE** 


Description	Sets or returns the noise rate of the FM modulation (noise shape wave only).	
Syntax	:AWG<x>:MODulation:FM:RATE {RATE10M   RATE5M   RATE1M   RATE500K   RATE100K   RATE50K   RATE10K   RATE5K   RATE1K   ?}	
Parameter/ Return parameter	<x>	Channel number 1~2.
	RATE10M	10MHz noise rate.
	RATE5M	5MHz noise rate.
	RATE1M	1MHz noise rate.
	RATE500K	500kHz noise rate.
	RATE100K	100kHz noise rate.
	RATE50K	50kHz noise rate.
	RATE10K	10kHz noise rate.
	RATE5K	5kHz noise rate.
	RATE1K	1kHz noise rate.

**Example** :AWG1:MOD:FM:RATE RATE5K  
 Sets the noise rate of the FM modulating waveform to 5kHz.

**:AWG<x>:MODulation:FSK:FREQ** 


Description	Sets or returns the hop frequency of the FSK modulation.	
Syntax	:AWG<x>:MODulation:FSK:FREQ {<NRf>   ?}	
Parameter/ Return parameter	<x>	Channel number 1~2.
	<NRf>	Frequency in Hertz.

**Example** :AWG1:MOD:FSK:FREQ 2000000  
 Sets the FSK hop frequency to 2MHz.

Set →  
 → Query

**:AWG<x>:MODulation:FSK:RATE**

Description	Sets or returns the FSK modulation rate.	
Syntax	:AWG<x>:MODulation:FSK:RATE {<NRf>   ?}	
Parameter/ Return parameter	<x>	Channel number 1~2.
	<NRf>	Frequency in Hertz.
Example	:AWG1:MOD:FSK:RATE 100000 Sets the FSK rate to 100kHz.	

Set →  
 → Query

**:AWG<x>:SWEep:STATE**

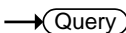
Description	Sets or returns the Sweep mode state.	
Syntax	:AWG<x>:SWEep:STATE {OFF   ON   ?}	
Parameter/ Return parameter	<x>	Channel number 1~2.
	OFF	Sets the sweep mode to off.
	ON	Sets the sweep mode to on.
Example	:AWG1:SWE:STATE ON Turns the sweep mode to on for channel 1.	

Set →  
 → Query

**:AWG<x>:SWEep:TYPe**

Description	Sets or returns the sweep mode type.	
Syntax	:AWG<x>:SWEep:TYPe {LINEAR   LOG   ?}	
Parameter/ Return parameter	<x>	Channel number 1~2.
	LINEAR	Sets the sweep mode to linear.
	LOG	Sets the sweep mode to logarithmic.
Example	:AWG1:SWE:TYP LIN Sets the sweep mode to linear for channel 1.	

**:AWG<x>:SWEep:START** 

**Description**      Sets or returns the start frequency of the sweep mode.

**Syntax**            :AWG<x>:SWEep:START {<NRf> | ?}

<b>Parameter/ Return parameter</b>	<b>&lt;x&gt;</b>	Channel number 1~2.
	<b>&lt;NRf&gt;</b>	Start frequency in Hertz.

**Example**            :AWG1:SWE:START 1000  
 Sets the sweep mode start frequency to 1kHz.

**:AWG<x>:SWEep:STOP** 


**Description**      Sets or returns the stop frequency of the sweep mode.

**Syntax**            :AWG<x>:SWEep:STOP {<NRf> | ?}

<b>Parameter/ Return parameter</b>	<b>&lt;x&gt;</b>	Channel number 1~2.
	<b>&lt;NRf&gt;</b>	Stop frequency in Hertz.

**Example**            :AWG1:SWE:STOP 500000  
 Sets the sweep mode stop frequency to 500kHz.

**:AWG<x>:SWEep:TIME** 


**Description**      Sets or returns the sweep time.

**Syntax**            :AWG<x>:SWEep:TIME {<NRf> | ?}

<b>Parameter/ Return parameter</b>	<b>&lt;x&gt;</b>	Channel number 1~2.
	<b>&lt;NRf&gt;</b>	Sweep time in seconds.

**Example**            :AWG1:SWE:TIM 6.500e-01  
 Sets the sweep time to 650ms.

Set →  
 → Query

**:AWG<x>:SWEep:SPAN**

**Description**      Alternatively to setting the start and stop frequencies, the span and center frequency can be set.

**Syntax**            :AWG<x>:SWEep:SPAN {<NRf> | ?}

<b>Parameter/ Return parameter</b>	<x>	Channel number 1~2.
	<NRf>	Span of the sweep in Hertz.

**Example**            :AWG1:SWE:SPAN 1100  
                          Sets the span of the sweep to 1.1kHz.

Set →  
 → Query

**:AWG<x>:SWEep:CENTer**

**Description**      Alternatively to setting the start and stop frequencies, the span and center frequency can be set.

**Syntax**            :AWG<x>:SWEep:CENTer {<NRf> | ?}

<b>Parameter/ Return parameter</b>	<x>	Channel number 1~2.
	<NRf>	Center frequency of the sweep in Hertz.

**Example**            :AWG1:SWE:CENT 550  
                          Sets the center frequency of the sweep to 550Hz.

Set →  
 → Query

**:AWG<x>:ARBITrary:EDIT:NUMPoint**

**Description**      Sets or returns the number of points of an arbitrary waveform.

**Syntax**            :AWG<x>:ARBITrary:EDIT:NUMPoint { <NR1> | ?}

<b>Parameter/ Return parameter</b>	<x>	Channel number 1~2.
	<NR1>	Number of points



Example :AWG1:ARB:EDIT:NUMP 1500  
Sets 1500 points for the arbitrary waveform.

**:AWG<x>:ARBitrary:EDIT:FUNCTion** (Set) →

Description	Sets the inbuilt waveform of the arbitrary waveform.	
Syntax	:AWG<x>:ARBitrary:EDIT:FUNCTion { SINE   SQUare   PULSe   RAMP   NOISe }	
Parameter	<x>	Channel number 1~2.
	SINE	Sine wave shape.
	SQUare	Square wave shape.
	PULSe	Pulse wave shape.
	RAMP	Ramp wave shape.
	NOISe	Noise wave shape.

Example :AWG1:ARB:EDIT:FUNCT RAMP  
Sets a ramp shape to the arbitrary waveform.

**:AWG<x>:ARBitrary:SAVe:WAVEform** (Set) →

Description	Saves an arbitrary waveform.	
Syntax	:AWG<x>:ARBitrary:SAVe:WAVEform {ARB1  ARB2   ARB3   ARB4   <file path>}	
Parameter	<x>	Channel number 1~2.
	ARB1~4	Saves the arbitrary waveform to one of the internal memory slots.
	<file path>	Saves the arbitrary waveform to disk or USB to the specified file path. Example: "Disk:/xxx.UAW" "USB:/xxx.UAW"

Example :AWG1:ARB:SAVE:WAVE ARB2  
Saves the arbitrary waveform to ARB2.

**:AWG<x>:ARBitrary:LOAd:WAVEform Set →**

Description	Loads an arbitrary waveform.	
Syntax	:AWG<x>:ARBitrary:LOAd:WAVEform { ARB1   ARB2   ARB3   ARB4   <file path> }	
Parameter	<x>	Channel number 1~2.
	ARB1~4	Loads the arbitrary waveform from one of the internal memory slots.
	<file path>	Loads the arbitrary waveform from disk or USB at the specified file path. Exemple: "Disk:/xxx.UAW" "USB:/xxx.UAW"

Example :AWG1:ARB:LOA:WAVE ARB2  
Loads the arbitrary waveform from ARB2.

**:AWG<x>:ARBitrary:EDIT:COPIY Set →**

Description	Copies a segment of an arbitrary waveform to a specific point.	
Syntax	:AWG<x>:ARBitrary:EDIT:COPIY {<START> , <LENGth> , <PASTe> }	
Parameter	<x>	Channel number 1~2.
	<START>	NR1, point at which the segment to copy starts.
	<LENGth>	NR1, length of the segment to copy.
	<PASTe>	NR1, point at which the segment is to be copied.

**Example** :AWG1:ARB:EDIT:COPY 5,100,106  
 Copies a segment of 100 points starting from point 5 of an arbitrary waveform and paste it to point 106 of this arbitrary waveform.

**:AWG<x>:ARBitrary:EDIT:CLEar** (Set) →

<b>Description</b>	Deletes a segment of an arbitrary waveform	
<b>Syntax</b>	:AWG<x>:ARBitrary:EDIT:CLEar { ALL   <START> , <LENGTh> }	
<b>Parameter</b>	<x>	Channel number 1~2.
	ALL	Deletes the entire arbitrary waveform.
	<START>	NR1, point at which the segment to delete starts.
	<LENGTh>	NR1, length of the segment to delete.

**Example** :AWG1:ARB:EDIT:CLE ALL

**:AWG<x>:ARBitrary:EDIT:LINE** (Set) →

<b>Description</b>	Creates a line on an arbitrary waveform.	
<b>Syntax</b>	:AWG<x>:ARBitrary:EDIT:LINE {<address1> , <data1> , address2> , <data2> }	
<b>Parameter</b>	<x>	Channel number 1~2.
	<address1>	NR1, the point at which the line starts.
	<data1>	NRf, the value at the starting point.
	<address2>	NR1, the point at which the line ends.
	<data2>	NRf, the value at the ending point.

Example :AWG1:ARB:EDIT:LIN 40,0.05,100,0.1  
 Creates a line between point 40 at value 0.05 and point 100 at value 0.01.

**:AWG<x>:ARBitrary:EDIT:SCALE Set →**

Description Sets the vertical scale of the arbitrary waveform.

Syntax :AWG<x>:ARBitrary:EDIT:SCALE {<NRf>}

Parameter	<x>	Channel number 1~2.
	<NRf>	Scale 0.1~ 10

Example :AWG1:ARB:EDIT:SCAL 5.5

**:AWG<x>:ARBitrary:EDIT:POINT Set →**

Description Edits a single point on an arbitrary waveform.

Syntax :AWG<x>:ARBitrary:EDIT:POINT {<address1> , <data1>}

Parameter	<x>	Channel number 1~2.
	<address1>	NR1, the point to be edited.
	<data1>	NRf, the value of that point.

Example :AWG1:ARB:EDIT:POIN 20,0.2

**:AWG<x>:ARBitrary:EDIT:POINT:ADD Set →**

Description Adds the edited point to the arbitrary waveform.

Syntax :AWG<x>:ARBitrary:EDIT:POINT:ADD {<NR1>}

Parameter	<x>	Channel number 1~2.
	<NR1>	The point to be added.

Example :AWG1:ARB:EDIT:POIN:ADD 20

:AWG<x>:ARbitrary:EDIT:POINT:DELEte 

---

Description Adds the edited point to the arbitrary waveform.

---

Syntax :AWG<x>:ARbitrary:EDIT:POINT:DELEte {<NR1>}

---

Parameter <x> Channel number 1~2.

<NR1> The point to be deleted.

---

Example :AWG1:ARB:EDIT:POIN:DELE 20

## Data Logging Commands

:DATALOG:STATE .....	326
:DATALOG:SOURce .....	326
:DATALOG:SAVe .....	327
:DATALOG:INTerval.....	327
:DATALOG:DURation.....	328

:DATALOG:STATE (Set) →  
→ (Query)

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 (Set) →  
→ (Query)

Description	Sets or queries the data logging source channel.	
Syntax	:DATALOG:SOURce { CH1~CH4   D0~D15   all   ? }	
Related commands	:DATALOG:STATE :DATALOG:SAVe :DATALOG:INTerval :DATALOG:DURation	

Parameter/Return parameter	CH1 ~CH4	Channel 1, 2, 3 or 4
	D0~D15	Digital channels D0~D15
	all	All displayed channels.

Example :DATALOG:SOURce CH1  
Sets the source to CH1.

Set →

→ Query

### :DATALOG:SAVe

Description Sets or queries the save format as image or waveform.

Syntax :DATALOG:SAVe {IMAGe|WAVEform|?}

Related commands :DATALOG:STATE  
:DATALOG:SOURce  
:DATALOG:INTerval  
:DATALOG:DURation

Parameter/Return parameter	IMAGe	Save as images.
	WAVEform	Save as waveforms.

Example :DATALOG:SAVe WAVEform  
Sets the save format to waveform.

Set →

→ Query

### :DATALOG:INTerval

Description Sets or queries the interval time between each recording.

Syntax :DATALOG:INTerval <NRF>  
:DATALOG:INTerval?

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

Parameter	<code>&lt;NRF&gt;</code>	Discrete time intervals in seconds:
-----------	--------------------------	-------------------------------------

Example	:DATALOG:INT 2 Sets the interval time to 2 seconds.
---------	--

(Set) →

→ (Query)

**:DATALOG:DURation**

Description	Sets or queries the duration time of each recording.
-------------	--

Syntax	:DATALOG:DURation <Nrf> :DATALOG:DURation?
--------	---

Related commands	:DATALOG:STATE :DATALOG:SOURce :DATALOG:SAVe :DATALOG:INTerval
------------------	---

Parameter	<code>&lt;Nrf&gt;</code>	Discrete recording time in seconds.
-----------	--------------------------	-------------------------------------

Example	:DATALOG:DUR 5 Sets the recording time to 5 seconds.
---------	---



## Remote Disk Commands

:REMOTEDisk:IPADdress .....	329
:REMOTEDisk:PATHName .....	329
:REMOTEDisk:USERName .....	329
:REMOTEDisk:PASSWord .....	330
:REMOTEDisk:MOUNT.....	330
:REMOTEDisk:AUTOMount.....	331

:REMOTEDisk:IPADdress 



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 



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 



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

Syntax	:REMOTEDisk:USERName {<string>   ? }	
Parameter/Return parameter	<string>	User name enclosed in double quotes eg., "User_Name".
Example	:REMOTEDisk:USERName "User_Name" Sets the account name as User_Name.	

Set →

→ Query

**:REMOTEDisk:PASSWord**


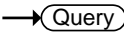
Description	Sets or queries the account password for the remote disk.	
Syntax	:REMOTEDisk:PASSWord {<string>   ? }	
Parameter/Return parameter	<string>	Username password enclosed in double quotes eg., "Password".
Example	:REMOTEDisk:PASSWord "Password" Sets the account password as Password.	

Set →

→ Query

**:REMOTEDisk:MOUNT**

Description	Turns remote disk on/off or queries its state.	
Syntax	:REMOTEDisk:MOUNT { OFF   ON   ? }	
Parameter/Return parameter	OFF	Unmount remote disk
	ON	Mount remote disk
Example	:REMOTEDisk:IPADDRESS "172.16.5.154" :REMOTEDisk:PATHName "remote_disk" :REMOTEDisk:USERName "guest" :REMOTEDisk:PASSWord "password" :REMOTEDisk:MOUNT ON Sets the remote disk parameters and mounts the remote disk.	

		 
<b>:REMOTEDisk:AUTOMount</b>		
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.	

## Spectrum Analyzer Commands

:SA:STATE .....	333
:SA:LIST .....	333
:SA:MEMory .....	333
:SA:MEMory:SOURce .....	335
:SA:SOURce .....	336
:SA<x>:SPECTRUMTrace .....	336
:SA<x>:NORMal .....	337
:SA<x>:MAXHold .....	337
:SA<x>:MINHold .....	337
:SA<x>:AVERage .....	338
:SA<x>:AVERage:NUMAVg .....	338
:SA<x>:DETECTIonmethod:MODE .....	339
:SA<x>:DETECTIonmethod:MAXHold .....	339
:SA<x>:DETECTIonmethod:MINHold .....	340
:SA<x>:DETECTIonmethod:NORMal .....	340
:SA<x>:DETECTIonmethod:AVERage .....	341
:SA<x>:FREQuency .....	341
:SA<x>:SPAN .....	342
:SA<x>:START .....	342
:SA<x>:STOP .....	342
:SA<x>:RBW:MODE .....	343
:SA<x>:RBW .....	343
:SA<x>:SPANRbwratio .....	344
:SA<x>:WINDow .....	344
:SA<x>:UNIts .....	345
:SA<x>:SCALE .....	345
:SA:POSition .....	346
:SA<x>:INPut .....	346
:SA:SPECTRogram:NUMSLICES? .....	347
:SA<x>:SPECTRogram:SLICeSElect .....	347
:SA<x>:SPECTRogram:SLICETIme? .....	347
:SA:SPECTRogram:STATE .....	348

**:SA:STATE**

Set →  
→ Query

Description	Sets or returns the state of the spectrum analyzer.	
Syntax	:SA:STATE {OFF ON} :SA:STATE?	
Parameter	OFF	Disable this function.
	ON	Enable this function.
Example	SA:STATE ON SA:STATE? ON	

**:SA:LIST**

→ Query

Description	Returns the data of the spectrum analyzer peak table.	
Syntax	:SA:LIST?	
Example	SA:LIST? NO., Frequency, Value; 1, 1.482E+07, -7.680E+01; 2, 2.790E+07, -7.600E+01; 3, 3.670E+07, -7.600E+01;	

**:SA:MEMory**

→ Query

Description	Returns the data in acquisition memory for the spectrum analyzer function as a header + raw data.	
Syntax	:SA:MEMory?	
Related Commands	:SA:MEMory:SOURce	
Return parameter	<string> <waveform block data>	Returns acquisition settings followed by raw waveform block data.
	<string>	

Returns the spectrum analyzer settings .

Format:

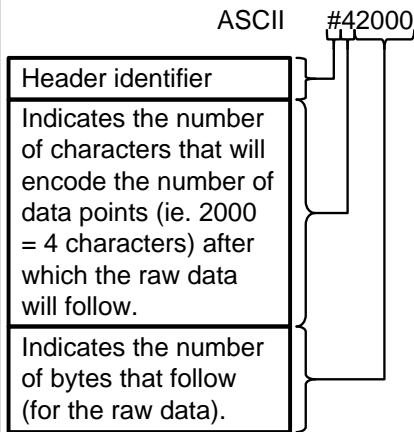
parameter(1),setting(1);parameter(2),setting(2)...parameter(n),setting(n);Waveform Data;

<waveform block data>

Header followed by the raw waveform data.

Format:

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



Raw Data:

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

Waveform Raw Data Example:

Header raw data.....

Hex:

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

1A 00 1B .....

ASCII/Decimal:

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

The actual value of a data point can be calculated with the following formula:

(Decimal value of hex data/AD Factor) \* vertical scale.

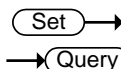
Note: AD Factor is fixed as 25. The vertical scale is returned with the acquisition settings that precede the raw data.

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

Example

:SA:MEMory?

Format,2.0E;Firmware,V1.28;Time,24-Apr-17  
 15:54:49;Memory  
 Length,1.000E+03;Source,CH1;Probe  
 Ratio,1.000E+00;Vertical Unit,dB;Vertical  
 Position,3.000E+00;Vertical  
 Scale,2.000E+01;Horizontal Unit,HZ;Horizontal  
 Scale,1.000E+04;Sampling Period,1.000E+02;Center  
 Frequency,2.300E+03;Span,1.000E+05;FREQUENCY,N  
 ORM,Waveform Data;  
 #42000 ..... follows waveform block data in  
 hex



:SA:MEMory:SOURce

Description	Sets or returns the source of the waveform data
Syntax	:SA:MEMory:SOURce {NORMAL   AVERAGE  MAXHold   MINHold} :SA:MEMory:SOURce?

Parameter/Return parameter	NORMAL	Normal data
	AVerage	Average data
	MAXHold	Maxhold data
	MINHold	Minhold data

Example :SA:MEMory:SOURce AVE  
Sets the memory source to average data.

:SA:SOURce (Set) →  
→ (Query)

Description Sets or returns the source of the spectrum analyzer

Syntax :SA:SOURce {CH1 | CH2| CH3 | CH4}  
:SA:SOURce?

Parameter/Return parameter	CH1	Channel one
	CH2e	Chnanel two
	CH3	Channel three
	CH4	Channel four

Example :SA:SOURce CH2  
Sets the source of spectrum analyzer to channel two.

:SA<x>:SPECTRUMTrace (Set) →

Description Resets all spectrum traces.

Syntax SA<x>:SPECTRUMTrace {RESET}

Parameter	RESET	Reset the trace
	<x>	1~2

Example :SA1:SPECTRUMTrace RESET  
Reset the trace one of spectrum analyzer.



**:SA<x>:NORMal** 


**Description** Sets or returns the frequency domain Normal trace display on or off in the frequency domain graticule.

**Syntax** :SA<x>:NORMal {ON|OFF}  
:SA<x>t:NORMal?

<b>Parameter/Return parameter</b>	ON	Turns the normal trace display on.
	OFF	Turns the normal trace display off.
	<x>	1~2

**Example** :SA<1>:NORMal ON  
Sets the normal trace one display on.

**:SA<x>:MAXHold** 


**Description** Sets or returns the frequency domain Max Hold trace display on or off in the frequency domain graticule.

**Syntax** :SA<x>:MAXHold {ON|OFF}  
:SA<x>:MAXHold?

<b>Parameter/Return parameter</b>	ON	Turns the Max Hold trace display on.
	OFF	Turns the Max Hold trace display off.
	<x>	1~2

**Example** :SA<1>:MAXHold OFF  
Sets the Max Hold trace one display off.

**:SA<x>:MINHold** 


**Description** Sets or returns the frequency domain Min Hold trace display on or off in the frequency domain graticule.

Syntax : SA<x>:MINHold {ON|OFF}  
 : SA<x>:MINHold?

Parameter/Return parameter	ON	Turns the Min Hold trace display on.
	OFF	Turns the Min Hold trace display off.
	<x>	1~2

Example : SA<2>:MINHold OFF  
 Sets the Min Hold trace two display off.

(Set) →

→ (Query)

**:SA<x>:AVERage**

Description Sets or returns the frequency domain Average trace display on or off in the frequency domain graticule.

Syntax : SA<x>:AVERage {ON|OFF}  
 : SA<x>: AVERage?

Parameter/Return parameter	ON	Turns the Average trace display on.
	OFF	Turns the Average trace display off.
	<x>	1~2

Example : SA<1>: AVERage ON  
 Sets the Average trace one display on.

(Set) →

→ (Query)

**:SA<x>:AVERage:NUMAVg**

Description Sets or returns the number of acquisitions to be used when creating the Average frequency domain trace.

Syntax :SA<x>:AVERage:NUMAVg {<NR1>}  
 :SA<x>:AVERage:NUMAVg?

Parameter/Return parameter	<NR1>	The range is 2 – 256, in exponential increments.
	<x>	1~2

Example :SA<1>:AVERage:NUMAVg 128  
Sets the Average number of trace one to 128.

:SA<x>:DETECTIonmethod:MODE   


Description Sets or returns the detection within the oscilloscope occurs automatically or manually.

Syntax :SA<x>:DETECTIonmethod:MODE {AUTO|MANual}  
:SA<x>:DETECTIonmethod:MODE?

Related commands :SA<x>:DETECTIonmethod:MAXHold,:SA:DETECTIonmethod:MINHold  
:SA<x>:DETECTIonmethod:NORMAL,:SA:DETECTIonmethod:AVERage

Parameter/Return parameter	AUTO	Automatically mode
	MANual	Manually mode
	<x>	1~2

Example :SA<1>:DETECTIonmethod:MODE AUTO  
Sets the detection mode of trace one to automatic.

:SA<x>:DETECTIonmethod:MAXHold   


Description Sets or returns the detection method of max Hold frequency domain trace.

Syntax :SA<x>:DETECTIonmethod:MAXHold {PLUSpeak|MINUSpeak|SAMPLE|AVERage}  
:SA<x>:DETECTIonmethod:MAXHold?

Parameter/Return parameter	PLUSpeak	Sets the detection method to plus peak.
	MINUpeak	Sets the detection method to minus peak.
	SAMPLE	Sets the detection method to sample.
	AVERage	Sets the detection method to average.

<x> 1~2

Example :SA<1>:DETECTIonmethod:MAXHold AVErAge  
Sets the detection method of trace one to average.

Set →

:SA<x>:DETECTIonmethod:MINHold → Query

Description Sets or returns the detection method of min Hold frequency domain trace.

Syntax :SA<x>:DETECTIonmethod:MINHold  
{PLUSpeak|MINUSpeak|SAMPLE|AVErAge}  
:SA<x>:DETECTIonmethod:MINHold?

Parameter/Return parameter	PLUSpeak	Sets the detection method to plus peak.
	MINUpeak	Sets the detection method to minus peak.
	SAMple	Sets the detection method to sample.
	AVErAge	Sets the detection method to average.
	<x>	1~2

Example :SA<2>:DETECTIonmethod:MINHold AVErAge  
Sets the detection method of trace two to average.

Set →

:SA<x>:DETECTIonmethod:NORMAL → Query

Description Sets or returns the detection method of normal frequency domain trace.

Syntax :SA<x>:DETECTIonmethod:NORMAL  
{PLUSpeak|MINUSpeak|SAMPLE|AVErAge}  
:SA<x>:DETECTIonmethod:NORMAL?

Parameter/Return parameter	PLUSpeak	Sets the detection method to plus peak.
	MINUpeak	Sets the detection method to minus peak.

SAMple	Sets the detection method to sample.
AVErage	Sets the detection method to average.
<x>	1~2

Example :SA<1>:DETECTIonmethod:NORMAL AVErage  
Sets the detection method of trace one to average.

:SA<x>:DETECTIonmethod:AVErage (Set) →  
→ (Query)

Description Sets or returns the detection method of average frequency domain trace.

Syntax :SA<x>:DETECTIonmethod:AVErage  
{PLUSpeak|MINUSpeak|SAMPlE|AVErage}  
:SA<x>:DETECTIonmethod:AVErage?

Parameter/Return parameter	PLUSpeak	Sets the detection method to plus peak.
	MINUpeak	Sets the detection method to minus peak.
	SAMple	Sets the detection method to sample.
	AVErage	Sets the detection method to average.
	<x>	1~2

Example :SA<1>:DETECTIonmethod:AVErage AVErage  
Sets the detection method of trace one to average.

:SA<x>:FREQuency (Set) →  
→ (Query)

Description Sets or returns the frequency (or center frequency) of the acquisition system.

Syntax :SA<x>:FREQuency {<NRF>|CENTER}  
:SA<x>:FREQuency?

Parameter/Return parameter	<NRF>	Sets the frequency by user.
	CENTER	Sets the frequency to center.

<x> 1~2

Example SA<1>:FREQuency 3.0E+06  
Sets the center frequency of trace one to 3 MHz.

Set →

:SA<x>:SPAN

→ Query

Description Sets or returns the span frequency setting.

Syntax :SA<x>:SPAN <Nrf>  
:SA<x>:SPAN?

Parameter/Return parameter <Nrf> Sets the span frequency by user.

Example SA<1>:SPAN 25E+06  
Sets the span frequency of trace one to 25 MH.

Set →

:SA<x>:START

→ Query

Description Sets or returns the start frequency setting.

Syntax :SA<x>:START <Nrf>  
:SA<x>:START?

Parameter/Return parameter <Nrf> Sets the start frequency by user.  
<x> 1~2

Example SA<1>:START -9.5E+06  
Sets the start frequency of trace one to -9.5 MHz.

Set →

:SA<x>:STOP

→ Query

Description Sets or returns the stop frequency setting.

Syntax :SA<x>:STOP <Nrf>  
:SA<x>:STOP?

Parameter/Return parameter <Nrf> Sets the stop frequency by user.  
<x> 1~2

Example            SA<1>:START 100E+06  
                      Sets the stop frequency of trace one to 100MHz.

Set →  
 → Query

Description       Sets or returns the resolution bandwidth (RBW) mode, either automatic or manual.

Syntax             :SA<x>:RBW:MODE {AUTO|MANual}  
                      :SA<x>:RBW:MODE?

Parameter/Return parameter	AUTo	Automatically mode.
	MANual	Manually mode.
	<x>	1~2

Example            SA<1>:RBW:MODE AUTo  
                      Sets the mode of trace one to automatic.

Set →  
 → Query

Description       Sets or returns the resolution bandwidth (RBW) when the RBW mode has been set to MANUAL (using the command SA:RBW:MODE).

Syntax             :SA<x>:RBW <NRf>  
                      :SA<x>:RBW?

Related commands    SA<x>:RBW:MODE

Parameter/Return parameter	<NRf>	Sets the RBW by user.
	<x>	1~2

Example            Sets SA<1>:RBW 2.0E+04  
                      Query SA<1>:RBW?  
                      Return 1.825017e+04  
                      If the RBW set to 20kHz, the query will return the nearestvalue (1.825017e+04).

Set →

→ Query

**:SA<x>:SPANRbwratio**

**Description**      Sets or returns the resolution bandwidth (RBW) when the RBW mode has been set to AUTO (using the command SA<x>:RBW:MODE).

**Syntax**            :SA<x>:SPANRbwratio  
 {RATIO1K|RATIO2K|RATIO5K|RATIO10K|RATIO20K|RATIO50K|RATIO100K|RATIO200K|<NRf>}  
 :SA<x>:SPANRbwratio?

**Related commands**      SA<x>:RBW:MODE

<b>Parameter/ Return parameter</b>	<NRf>	Sets the RBW by user.
	RATIO1K	1000 : 1
	RATIO2K	2000 : 1
	RATIO5K	5000 : 1
	RATIO10K	10000 : 1
	RATIO20K	20000 : 1
	RATIO50K	50000 : 1
	RATIO100K	100000 : 1
	RATIO200K	200000 : 1
	<x>	1~2

**Example**            :SA<1>:SPANRbwratio RATIO2K  
 Sets the ratio of trace one to 2000:1.  
 Sets :SA<1>:SPANRbwratio 2000  
 Query :SA<1>:SPANRbwratio?  
 Return RATIO2K

Set →

→ Query

**:SA<x>:WINDow**

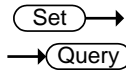
**Description**      Sets or returns the windowing function, which is only used for traces.



Syntax :SA<x>:WINDow {RECTangular|HAMming|HANning|BLAckman}  
:SA<x>:WINDow?

Parameter/ Return parameter	RECTangular	Sets to Rectangular window
	HAMming	Sets to Hamming window
	HANning	Sets to Hanning window
	BLAckman	Sets to Blackman window
	<x>	1~2

Example :SA<1>:WINDow HANning  
Sets to the hanning window for the trace one.



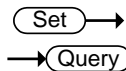
**:SA<x>:UNIIts**

Description Sets or returns the vertical units.

Syntax :SA<x>:UNIIts {DBV|LINEAR|DBM}  
:SA<x>:UNIIts?

Parameter/ Return parameter	DBV	Sets to DBV unit
	LINEAR	Sets to Linear unit
	DBM	Sets to DBM unit
	<x>	1~2

Example :SA<1>:UNIIts DBM  
Sets the unit of trace one to DBM unit.



**:SA<x>:SCALE**

Description Sets or returns the overall vertical scale.

Syntax :SA<x>:SCALE <NRF>  
:SA<x>:SCALE?

Related commands :SA<x>:UNIIts

Parameter/ Return parameter	<NRf>          <x>	Vertical scale, the value may vary which depends on the unit selected.  dBm and dBV :  1, 2, 5, 10, 20 (dB)  Linear:  2m, 5m, 10m, 20m, 50m, 100m, 200m, 500m, 1, 2, 5, 10, 20, 50, 100, 200, 500, 1k (V)  1~2
--------------------------------	--	--

Example :SA<1>:SCAlE 2  
Sets the scale of trace one to 2.

:SA:POSition (Set) →  
→ (Query)

Description	Sets or returns the overall vertical position.	
Syntax	:SA<x>:POSition <NRf> :SA<x>:POSition?	

Parameter/ Return parameter	<NRf>    <x>	Vertical position range: +/-12  1~2
--------------------------------	--------------------------	---

Example :SA<1>:POSition 3  
Sets the vertical position of trace one to 3.

:SA<x>:INPut (Set) →  
→ (Query)

Description	Sets or returns the input state of the spectrum analyzers.	
Syntax	:SA<x>:INPut {OFF ON} :SA<x>:INPut?	

Parameter	ON	Turn off the SA<x> input.
	OFF	Turn on the SA<x> input.
	<x>	1~2

Example SA2:INPut ON  
SA2:INPut?  
ON

**:SA:SPECTRogram:NUMSLICES?** → Query

Description Query the total number of slice in spectrogram.

Syntax :SA:SPECTRogram:NUMSLICES?

Example SA:SPECTRogram:NUMSLICES?  
57

Set →  
→ Query

**:SA<x>:SPECTRogram:SLICESElect**

Description Set or query the selected slice in spectrogram.

Syntax :SA<x>:SPECTRogram:SLICESElect <NR1>  
:SA<x>:SPECTRogram:SLICESElect?

Parameter <NR1> The number of slice. Range:0 ~ -197  
<x> 1~2

Example SA2:SPECTRogram:SLICESElect -20  
SA2:SPECTRogram:SLICESElect?  
-20

**:SA<x>:SPECTRogram:SLICETIME?** → Query

Description Query the timestamp of selected slice.

Syntax :SA<x>:SPECTRogram:SLICETIME?

Parameter <x> 1~2

Example SA1:SPECTRogram:SLICETIME?

Set →

→ Query

**:SA:SPECTRogram:STATe**

---

Description	Set or query the state of spectrogram display.	
Syntax	:SA:SPECTRogram:STATe {OFF ON} :SA:SPECTRogram:STATe?	
Parameter	OFF	Turn off the spectrogram display.
	ON	Turn on the spectrogram display.
Example	:SA:SPECTRogram:STATe ON :SA:SPECTRogram:STATe?	

## Power Analysis Commands

:POWer:TYPE .....	355
:POWer:CURRent:SOURce.....	355
:POWer:VOLTage:SOURce.....	356
:POWer:HARMonics:STANDard.....	356
:POWer:HARMonics:NR_HARMonics .....	356
:POWer:HARMonics:SOURce .....	357
:POWer:HARMonics:FREQRef.....	357
:POWer:HARMonics:FREQRef:FIXEDFREQValue...358	
:POWer:HARMonics:DISPlay:SElect.....	358
:POWer:HARMonics:DISPlay:TYPe.....	359
:POWer:HARMonics:IEC:LINEFREQuency.....	359
:POWer:HARMonics:IEC:OBSPERiod.....	359
:POWer:HARMonics:IEC:CLAss.....	360
:POWer:HARMonics:IEC:POWERFACTOR .....	360
:POWer:HARMonics:IEC:FUNDamental .....	360
:POWer:HARMonics:IEC:INPUTPOWer .....	361
:POWer:HARMonics:IEC:FILter .....	361
:POWer:HARMonics:IEC:GROUPing.....	361
:POWer:HARMonics:RESults:HAR<1-40>	
:REQuency? .....	362
:POWer:HARMonics:RESults:HAR<1-40>:ECMAX? 362	
:POWer:HARMonics:RESults:HAR<1-40>:LIMit? ...362	
:POWer:HARMonics:RESults:HAR<1-40>:HASe? .363	
:POWer:HARMonics:RESults:HAR<1-40>:RMS	
:ABSolute? .....	363
:POWer:HARMonics:RESults:HAR<1-40>:RMS	
:PERCent? .....	363
:POWer:HARMonics:RESults:HAR<1-40>:TEST	
:IEC:CLASSALIMit? .....	364
:POWer:HARMonics:RESults:HAR<1-40>:TEST	
:IEC:NORMAL? .....	364
:POWer:HARMonics:RESults:HAR<1-40>:TEST	
:IEC:POHCLIMit? .....	364

:POWer:HARMonics:RESuLts:IEC:FUNDamental? .365  
 :POWer:HARMonics:RESuLts:IEC  
 :HARM3ALTernate? .....365  
 :POWer:HARMonics:RESuLts:IEC  
 :HARM5ALTernate? .....366  
 :POWer:HARMonics:RESuLts:IEC:POHC? .....366  
 :POWer:HARMonics:RESuLts:IEC:POWer? .....366  
 :POWer:HARMonics:RESuLts:IEC:POWERFactor? ..366  
 :POWer:HARMonics:RESuLts:PASSFail? .....367  
 :POWer:HARMonics:RESuLts:RMS? .....367  
 :POWer:HARMonics:RESuLts:THDF? .....367  
 :POWer:HARMonics:RESuLts:THDR? .....368  
 :POWer:HARMonics:RESuLts:SAVe .....368  
 :POWer:GATing .....368  
 :POWer:QUALity:DISPlay .....368  
 :POWer:QUALity:FREQREference .....369  
 :POWer:QUALity:VMAX? .....369  
 :POWer:QUALity:VMIN? .....370  
 :POWer:QUALity:IMAX? .....370  
 :POWer:QUALity:IMIN? .....370  
 :POWer:QUALity:DCVOLTage? .....370  
 :POWer:QUALity:DCCURRent? .....371  
 :POWer:QUALity:VCRESTfactor? .....371  
 :POWer:QUALity:ICRESTfactor? .....371  
 :POWer:QUALity:IMPedance? .....371  
 :POWer:QUALity:RESIstance? .....372  
 :POWer:QUALity:REACTance? .....372  
 :POWer:QUALity:APPpwr? .....372  
 :POWer:QUALity:FREQuency? .....372  
 :POWer:QUALity:ICRESTfactor? .....373  
 :POWer:QUALity:IRMS? .....373  
 :POWer:QUALity:PHASEangle? .....373  
 :POWer:QUALity:POWERFACTOR? .....373  
 :POWer:QUALity:REACTpwr? .....374  
 :POWer:QUALity:TRUEpwr? .....374

:POWER:QUALity:VRMS?	374
:POWER:RIPple	374
:POWER:RIPple:RESults:AMPlitude?	375
:POWER:RIPple:RESults:REALAmplitude?	375
:POWER:RIPple:SOURce	375
:POWER:INRUshcurrent:RESults:FIRStpeak?	375
:POWER:INRUshcurrent:RESults:SECondpeak?	376
:POWER:MODulation:SOURce	376
:POWER:MODulation:TYPe	376
:POWER:REFLevel:PERCent:MID	377
:POWER:SOA:AXESTYPe	377
:POWER:SOA:LINear:XMAX	377
:POWER:SOA:LINear:XMIN	378
:POWER:SOA:LINear:YMAX	378
:POWER:SOA:LINear:YMIN	378
:POWER:SOA:LOG:XMAX	379
:POWER:SOA:LOG:XMIN	379
:POWER:SOA:LOG:YMAX	379
:POWER:SOA:LOG:YMIN	380
:POWER:SOA:MASK:TYPe	380
:POWER:SOA:MASK:NUMPt	380
:POWER:SOA:MASK:POINt<x>	381
:POWER:SOA:MASK:MAXVoltage	381
:POWER:SOA:MASK:MAXCURRent	381
:POWER:SOA:MASK:MAXPower	382
:POWER:SOA:MASK:STOPOnviol	382
:POWER:SOA:RESults:FAILures?	382
:POWER:SOA:RESults:NUMACq?	382
:POWER:SOA:MASK:RESults:STATe?	383
:POWER:SWLoss:CONDCALCtype	383
:POWER:SWLoss:DISPlay	383
:POWER:SWLoss:GATing	384
:POWER:SWLoss:RDSon	384
:POWER:SWLoss:VCEsat	384
:POWER:SWLoss:REFLevel:PERCent:MID	385

:POWer:SWLoss:REFLevel:PERCent:CURRent ..... 385  
 :POWer:SWLoss:REFLevel:PERCent:VOLTAge ..... 385  
 :POWer:SWLoss:CONDUction:ENERgy:MAX? ..... 385  
 :POWer:SWLoss:CONDUction:ENERgy:MEAN? ..... 386  
 :POWer:SWLoss:CONDUction:ENERgy:MIN? ..... 386  
 :POWer:SWLoss:CONDUction:POWer:MAX? ..... 386  
 :POWer:SWLoss:CONDUction:POWer:MEAN? ..... 386  
 :POWer:SWLoss:CONDUction:POWer:MIN? ..... 386  
 :POWer:SWLoss:TOFF:ENERgy:MAX? ..... 387  
 :POWer:SWLoss:TOFF:ENERgy:MEAN? ..... 387  
 :POWer:SWLoss:TOFF:ENERgy:MIN? ..... 387  
 :POWer:SWLoss:TOFF:POWer:MAX? ..... 387  
 :POWer:SWLoss:TOFF:POWer:MEAN? ..... 387  
 :POWer:SWLoss:TOFF:POWer:MIN? ..... 388  
 :POWer:SWLoss:TON:ENERgy:MAX? ..... 388  
 :POWer:SWLoss:TON:ENERgy:MEAN? ..... 388  
 :POWer:SWLoss:TON:ENERgy:MIN? ..... 388  
 :POWer:SWLoss:TON:POWer:MAX? ..... 388  
 :POWer:SWLoss:TON:POWer:MEAN? ..... 389  
 :POWer:SWLoss:TON:POWer:MIN? ..... 389  
 :POWer:SWLoss:TOTAL:ENERgy:MAX? ..... 389  
 :POWer:SWLoss:TOTAL:ENERgy:MEAN? ..... 389  
 :POWer:SWLoss:TOTAL:ENERgy:MIN? ..... 389  
 :POWer:SWLoss:TOTAL:POWer:MAX? ..... 390  
 :POWer:SWLoss:TOTAL:POWer:MEAN? ..... 390  
 :POWer:SWLoss:TOTAL:POWer:MIN? ..... 390  
 :POWer:TRANSient:DURATION ..... 390  
 :POWer:TRANSient:OVERSHoot ..... 390  
 :POWer:TRANSient:STEADYVout ..... 391  
 :POWer:TRANSient:LOWCurrent ..... 391  
 :POWer:TRANSient:HIGHCurrent ..... 391  
 :POWer:TRANSient:RESults:VALue? ..... 392  
 :POWer:EFFiciency:STATIStics:MODE ..... 392  
 :POWer:EFFiciency:STATIStics:WEIghting ..... 392  
 :POWer:EFFiciency:STATIStics ..... 392



:POWer:EFFiciency:INPUTPOWer:VALue? .....393  
 :POWer:EFFiciency:INPUTPOWer:MEAN?.....393  
 :POWer:EFFiciency:INPUTPOWer:MINImum? .....393  
 :POWer:EFFiciency:INPUTPOWer:MAXmum?.....393  
 :POWer:EFFiciency:INPUTPOWer:STDdev?.....393  
 :POWer:EFFiciency:OUTPUTPOWer:VALue? .....394  
 :POWer:EFFiciency:OUTPUTPOWer:MEAN?.....394  
 :POWer:EFFiciency:OUTPUTPOWer:MINImum? ...394  
 :POWer:EFFiciency:OUTPUTPOWer:MAXmum? ....394  
 :POWer:EFFiciency:OUTPUTPOWer:STDdev? .....394  
 :POWer:EFFiciency:RESults:VALue? .....395  
 :POWer:EFFiciency:RESults:MEAN? .....395  
 :POWer:EFFiciency:RESults:MINImum? .....395  
 :POWer:EFFiciency:RESults:MAXmum? .....395  
 :POWer:EFFiciency:RESults:STDdev? .....395  
 :POWer:BHCurve:WINDings .....396  
 :POWer:BHCurve:CROSECArea .....396  
 :POWer:BHCurve:MAGNLength .....396  
 :POWer:BHCurve:B:POStion .....397  
 :POWer:BHCurve:B:SCAle .....397  
 :POWer:BHCurve:H:POStion .....397  
 :POWer:BHCurve:H:SCAle .....397  
 :POWer:TONOff:TEST .....398  
 :POWer:TONOff:SETup:DURation .....398  
 :POWer:TONOff:SETup:APPLY .....398  
 :POWer:TONOff:SETup:SAVESetup .....399  
 :POWer:TONOff:DISPlay .....399  
 :POWer:SHOrt:DISPlay .....399  
 :POWer:SHOrt:APPLY.....400  
 :POWer:OCP:DISPlay .....400  
 :POWer:OCP:APPLY.....400  
 :POWer:PSRR:RUN .....400  
 :POWer:PSRR:STOP .....401  
 :POWer:PSRR:SOURce:INPut .....401  
 :POWer:PSRR:SOURce:OUTPut .....401

:POWer:PSRR:FREQuency:STARt .....	401
:POWer:PSRR:FREQuency:STOP .....	402
:POWer:PSRR:AWG:LOAD.....	402
:POWer:PSRR:AWG:AMPlitude.....	402
:POWer:PSRR:POINt .....	403
:POWer:PSRR:SAVe .....	404
:POWer:PSRR:RECALL.....	404
:POWer:PSRR:DATA .....	404
:POWer:PSRR:SAVETOCsv.....	405
:POWer:PSRR:AWG:AMPlitude:PROFile .....	405
:POWer:PSRR:AWG:AMPlitude:INTERPolation .....	405
:POWer:PSRR:DATA:GMARgin .....	406
:POWer:PSRR:DATA:GMARgin:FREQuency.....	406
:POWer:PSRR:DATA:PMARgin .....	406
:POWer:PSRR:DATA:PMARgin:FREQuency .....	407
:POWer:CLR:RUN .....	407
:POWer:CLR:STOP .....	407
:POWer:CLR:SOURce:INPut .....	407
:POWer:CLR:SOURce:OUTPut .....	408
:POWer:CLR:FREQuency:STARt.....	408
:POWer:CLR:FREQuency:STOP .....	408
:POWer:CLR:AWG:LOAD.....	409
:POWer:CLR:AWG:AMPlitude .....	409
:POWer:CLR:POINt .....	410
:POWer:CLR:SAVe .....	410
:POWer:CLR:RECALL.....	410
:POWer:CLR:DATA .....	411
:POWer:CLR:SAVETOCsv .....	411
:POWer:CLR:AWG:AMPlitude:PROFile.....	411
:POWer:CLR:AWG:AMPlitude:INTERPolation.....	412
:POWer:CLResponse:DATA:GMARgin .....	412
:POWer:CLResponse:DATA:GMARgin:FREQuency .....	413
:POWer:CLResponse:DATA:PMARgin .....	413
:POWer:PSRR:DATA:PMARgin:FREQuency .....	413

**:POWer:TYPe** 
 →  
 →

Description	Selects or returns the selected power analysis function.	
Syntax	:POWer:TYPe {NONE QUALity SWLoss HARMonics RIPPLE INRUshcurrent  MODulation SOA TRANSient EFFiciency BHCurve CLResponse PSRR TONOff   ?}	
Parameter/Return parameter	QUALity	Power quality function
	HARMonics	Harmonics function
	RIPPLE	Ripple function
	INRUshcurrent	Inrush current function
	SWLoss	Switching loss function
	MODulation	Modulation function
	SOA	Safe operation area function
	TRANSient	Transient function
	EFFiciency	Efficiency function
	BHCurve	B-H curve function
	CLResponse	Close loop response function
	PSRR	PSRR function
	TONOff	Turn ON/Off function

Example      :POWer:TYPe QUALity  
 Sets the power analysis function to power quality.

**:POWer:CURRent:SOURce** 
 →  
 →

Description	Sets or queries the current source.	
Syntax	:POWer:CURRent:SOURce {CH1 CH2 CH3 CH4 ?}	
Parameter/Return parameter	CH1~CH4	Channel of the current source.

Example :POWer:CURRent:SOURce CH1  
Sets the current source to CH1.

Set →

:POWer:VOLTage:SOURce

→ Query

Description Sets or queries the voltage source.

Syntax :POWer:VOLTage:SOURce {CH1|CH2|CH3|CH4 |?}

Parameter/Return parameter	CH1~CH4	Channel of the voltage source.
----------------------------	---------	--------------------------------

Example :POWer:VOLTage:SOURce CH2  
Sets the voltage source to CH2.

Set →

:POWer:HARMonics:STANDard

→ Query

Description Sets the harmonics standard to none or to IEC standards.

Syntax :POWer:HARMonics:STANDard {NONE|IEC |?}

Parameter/Return parameter	NONE	No harmonics standard.
	IEC	IEC standards

Example :POWer:HARMonics:STANDard NONE  
Sets the harmonics standard to none.

Set →

:POWer:HARMonics:NR\_HARMonics

→ Query

Description Sets the number of harmonics when the harmonic standard is set to none.

Note Only applicable if Standard is set to None.

Syntax :POWer:HARMonics:NR\_HARMonics {<NR3> | ?}

Parameter/Return parameter	<NRf>	20~400
	<NR3>	

Example :POWer:HARMonics:NR\_HARMonics 20  
Sets the number of harmonics to 20.

Set →

:POWer:HARMonics:SOURce

→ Query

Description Sets or queries the harmonics source when the harmonic standard is set to none.

Note Only CURRent is supported when the standard is set to IEC.

Syntax :POWer:HARMonics:SOURce {VOLTage|CURRent|?}

Parameter/Return parameter	VOLTage	Voltage source.
	CURRent	Current source.

Example :POWer:HARMonics:SOURce VOLTage  
Sets the harmonics source as the voltage source.

Set →

:POWer:HARMonics:FREQRef

→ Query

Description Sets or queries the harmonics reference when the harmonic standard is set to none.

Syntax :POWer:HARMonics:FREQRef {VOLTage | CURRent | HARMSOURce | FIXEDFREQuency | ?}

Parameter/Return parameter	VOLTage	Voltage source.
	CURRent	Current source.
	HARMSOURce	Harmonic source.
	FIXEDFREQuency	A fixed frequency value. The frequency is set by the FREQRef:FIXEDFREQValue command.

Example :POWer:HARMonics:FREQRef VOLTage  
Sets the harmonics reference as the voltage source.

Set →  
 → Query

**:POWer:HARMonics:FREQRef:FIXEDFREQValue**

Description	Sets or queries the fixed frequency value for the :POWer:HARMonics:FREQRef command.	
Note	This command is only applicable when the Standard is set to None and the frequency reference is set to Fixed.	
Syntax	:POWer:HARMonics:FREQRef:FIXEDFREQValue {<NR3>   ?}	
Parameter/Return parameter	<NRf> <NR3>	(10Hz to 400Hz)
Example	:POWer:HARMonics:FREQRef:FIXEDFREQValue 1.0E+1  Sets the fixed frequency to 10Hz.	

Set →  
 → Query

**:POWer:HARMonics:DISPlay:SElect**

Description	Sets or queries the whether the odd, even or all the harmonics are displayed in the results. This command is only applicable when the harmonic standard is set to none.	
Syntax	:POWer:HARMonics:DISPlay:SElect {ODD   EVEN   ALL   ?}	
Parameter/Return parameter	ODD EVEN ALL	Display only odd harmonics Display only even harmonics Display all the harmonics
Example	:POWer:HARMonics:DISPlay:SElect ODD  Display only the odd harmonics.	

**:POWer:HARMonics:DISPlay:TYPe** 


Description	Sets or queries the whether the results are displayed as a graph or as a table.	
Syntax	:POWer:HARMonics:DISPlay:TYPe {GRAph TABle ?}	
Parameter/Return parameter	GRAph	Display as graph
	TABle	Display as table
Example	:POWer:HARMonics:DISPlay:TYPe GRAph Display results in a graph.	

**:POWer:HARMonics:IEC:LINEFREQuency** 


Description	Sets or queries the line frequency when the harmonics standard is set to IEC.	
Syntax	:POWer:HARMonics:IEC:LINEFREQuency {<NR3>(50,60)   ?}	
Parameter/Return parameter	<NR3>	50 or 60.
Example	:POWer:HARMonics:IEC:LINEFREQuency 5.0e+1 Sets the line frequency to 50Hz.	

**:POWer:HARMonics:IEC:OBSPERiod** 


Description	Sets or queries the “observation period” in seconds when the harmonics standard is set to IEC.	
Syntax	:POWer:HARMonics:IEC:OBSPERiod {<NR3>(0.2~150)   ?}	
Parameter/Return parameter	<NRf>	0.2~150 seconds.
	<NR3>	
Example	:POWer:HARMonics:IEC:OBSPERiod 1.5E+2 Sets the observation period to 150 seconds.	

:POWer:HARMonics:IEC:CLAss (Set) →  
→ (Query)

Description	Sets or queries the IEC device class.			
Syntax	:POWer:HARMonics:IEC:CLAss {A B C1 C2 C3 D ?}			
Parameter/Return parameter	A	Class A	B	Class B
	C1	Class C (table1)	C2	Class C (table2)
	C3	Class C (table3)	D	Class D
Example	:POWer:HARMonics:IEC:CLAss B Sets the device class to B.			

:POWer:HARMonics:IEC:POWERFACTOR (Set) →  
→ (Query)

Description	Sets or queries the power factor when the class is set to C. This is only applicable when the harmonics standard is set to IEC.		
Syntax	:POWer:HARMonics:IEC:POWERFACTOR {<NR3>   ?}		
Parameter/Return parameter	<NRf> <NR3>	0.00~1.00	
Example	:POWer:HARMonics:IEC:POWERFACTOR 5.1E-1 Sets the power factor to 0.51.		

:POWer:HARMonics:IEC:FUNDamental (Set) →  
→ (Query)

Description	Sets or queries the class C current. This is only applicable when the harmonics standard is set to IEC		
Syntax	:POWer:HARMonics:IEC:FUNDamental {<NR3>   ?}		
Parameter/Return parameter	<NR3>	0.1A ~ 16A	
Example	:POWer:HARMonics:IEC:FUNDamental 1.5E+0 Sets the class C current to 1.5A.		



**:POWer:HARMonics:IEC:INPUtPOWer** 


**Description** Sets or queries the class C3/D input power. This is only applicable when the harmonics standard is set to IEC

**Syntax** :POWer:HARMonics:IEC:INPUtPOWer {<NR3> | ?}

**Parameter/Return parameter** <NRf> 0~600Watts. (10W steps)  
 <NR3>

**Example** :POWer:HARMonics:IEC:INPUtPOWer 1.0E+2  
 Sets the class C3/D input power to 100W.

**:POWer:HARMonics:IEC:FIlter** 


**Description** Turns the IEC harmonics filter on/off.

**Syntax** :POWer:HARMonics:IEC:FIlter {OFF | ON | ?}

**Parameter/Return parameter** OFF Filter off  
 ON Filter on

**Example** :POWer:HARMonics:IEC:FIlter ON  
 Turn the IEC harmonic filter on.

**:POWer:HARMonics:IEC:GRouPIng** 


**Description** Turns the IEC grouping on/off.

**Syntax** :POWer:HARMonics:IEC:GRouPIng {OFF | ON | ?}

**Parameter/Return parameter** OFF Grouping off  
 ON Grouping on

**Example** :POWer:HARMonics:IEC:GRouPIng ON  
 Turn grouping on.

:POWer:HARMonics:RESults:HAR<1-400>

:REQuency?

→ **Query**

Description Returns the frequency at the specified harmonic.

Syntax :POWer:HARMonics:RESults:HAR<1-400>  
:FREQuency?

Parameter <1-400> <NR1> Harmonic number.

Return parameter <NR3> Unit = Hz.

Example :POWer:HARMonics:RESults:HAR20:FREQuency?

:POWer:HARMonics:RESults:HAR<1-40>:ECMAX? → **Query**

Description Returns the “Max all Windows” result when the harmonics standard is set to IEC.

Syntax :POWer:HARMonics:RESults:HAR<1-40>:IECMAX?

Parameter <1-40> <NR1> Harmonic number.

Return parameter <NR3> Unit = A.

Example :POWer:HARMonics:RESults:HAR2:IECMAX?

:POWer:HARMonics:RESults:HAR<1-40>:LIMit? → **Query**

Description Returns the “Limit” result when the harmonics standard is set to IEC.

Syntax :POWer:HARMonics:RESults:HAR<1-40>:LIMit?

Parameter <1-40> <NR1> Harmonic number.

Return parameter <NR3> For device class C2, Unit = %.  
All other device classes, Unit = A.

Example :POWer:HARMonics:RESults:HAR1:LIMit?

:POWer:HARMonics:RESults:HAR<1-400>:HASe? → [Query](#)

Description	Returns the "Phase" (°) result when the harmonics standard is set to none.	
Syntax	:POWer:HARMonics:RESults:HAR<1-400>:PHASe?	
Parameter	<1-400>	<NR1> Harmonic number.
Return parameter	<NR3>	Unit = °.
Example	:POWer:HARMonics:RESults:HAR20:PHASe?	

:POWer:HARMonics:RESults:HAR<1-400>:RMS  
:ABSolute? → [Query](#)

Description	Returns the absolute "RMS" result.	
Syntax	:POWer:HARMonics:RESults:HAR1:RMS:ABSolute?	
Parameter	<1-400>	<NR1> Harmonic number.
Return parameter	<NR3>	Unit = V.
Example	:POWer:HARMonics:RESults:HAR20:RMS:ABSolute?	

:POWer:HARMonics:RESults:HAR<1-400>:RMS  
:PERCent? → [Query](#)

Description	Returns the "Mag%" result.	
Syntax	:POWer:HARMonics:RESults:HAR<1-400>:RMS :PERCent?	
Parameter	<1-400>	<NR1> Harmonic number.
Return parameter	<NR3>	Unit = %.
Example	:POWer:HARMonics:RESults:HAR20:RMS:PERCent?	

:POWER:HARMonics:RESults:HAR<1-40>:TEST  
:IEC:CLASSALIMit?

→ Query

Description Returns the class A “limit” result when the testing standard is set to IEC.

Syntax :POWER:HARMonics:RESults:HAR<1-40>:TEST:IEC  
:CLASSALIMit?

Parameter <1-40> <NR1> Harmonic number.

Return parameter	PASS	Passed limit testing
	FAIL	Failed limit testing
	N/A	N/A - device class is not A.

Example :POWER:HARMonics:RESults:HAR1:TEST:IEC  
:CLASSALIMit?

:POWER:HARMonics:RESults:HAR<1-40>:TEST  
:IEC:NORMAL?

→ Query

Description Returns the “limit” result for all device classes excluding class A. Only applicable for IEC.

Syntax :POWER:HARMonics:RESults:HAR<1-40>:TEST:IEC  
:NORMAL?

Parameter <1-40> <NR1> Harmonic number.

Return parameter	PASS	Passed limit testing
	FAIL	Failed limit testing

Example :POWER:HARMonics:RESults:HAR1:TEST:IEC  
:NORMAL?

:POWER:HARMonics:RESults:HAR<1-40>:TEST  
:IEC:POHCLIMit?

→ Query

Description Returns the “POHC Limit” result for all device classes when the standard is set to IEC.

Syntax	:POWER:HARMonics:RESults:HAR<1-40>:TEST:IEC :POHCLIMit?	
Parameter	<1-40>	<NR1> Harmonic number.
Return parameter	PASS FAIL NA	Passed limit testing Failed limit testing Not applicable
Example	:POWER:HARMonics:RESults:HAR1:TEST:IEC :POHCLIMit?	

**:POWER:HARMonics:RESults:IEC:FUNDamental? → Query**

Description	Returns the current level of the fundamental frequency. Only applicable with IEC.	
Syntax	:POWER:HARMonics:RESults:IEC:FUNDamental?	
Return parameter	<NR3>	Unit = A
Example	:POWER:HARMonics:RESults:IEC:FUNDamental?	

**:POWER:HARMonics:RESults:IEC  
:HARM3ALternate? → Query**

Description	Returns the limit test result of the IEC harmonic test for the 3 <sup>rd</sup> harmonic.	
Syntax	:POWER:HARMonics:RESults:IEC:HARM3ALternate?	
Return parameter	PASS FAIL NA	Passed limit testing Failed limit testing Not applicable
Example	:POWER:HARMonics:RESults:IEC:HARM3ALternate?	

:POWER:HARMonics:RESults:IEC

:HARM5ALTernate?

→ **Query**

**Description** Returns the limit test result of the IEC harmonic test for the 3<sup>rd</sup> harmonic.

**Syntax** :POWER:HARMonics:RESults:IEC:HARM5ALTernate?

<b>Return parameter</b>	PASS	Passed limit testing
	FAIL	Failed limit testing
	NA	Not applicable

**Example** :POWER:HARMonics:RESults:IEC:HARM5ALTernate?

:POWER:HARMonics:RESults:IEC:POHC?

→ **Query**

**Description** Returns the POHC measurement when the standard is set to IEC.

**Syntax** :POWER:HARMonics:RESults:IEC:POHC?

**Return parameter** <NR3> Unit = A

**Example** :POWER:HARMonics:RESults:IEC:POHC?

:POWER:HARMonics:RESults:IEC:POWER?

→ **Query**

**Description** Returns the input power for IEC device class C3.

**Syntax** :POWER:HARMonics:RESults:IEC:POWER?

**Return parameter** <NR3> Unit = W

**Example** :POWER:HARMonics:RESults:IEC:POWER?

:POWER:HARMonics:RESults:IEC:POWERFactor?

→ **Query**

**Description** Returns the power factor for IEC device classes C1, C2 and C3.

**Syntax** :POWER:HARMonics:RESults:IEC:POWERFactor?

Return parameter	<NR3>	0~1
------------------	-------	-----

Example :POWer:HARMonics:RESults:IEC:POWERFactor?

:POWer:HARMonics:RESults:PASSFail? → Query

Description	Returns the overall pass/fail result. Only applicable to when the standard is set to IEC.
-------------	---

Syntax	:POWer:HARMonics:RESults:PASSFail?
--------	------------------------------------

Return parameter	PASS	Passed limit testing
	FAIL	Failed limit testing
	NA	Not applicable

Example :POWer:HARMonics:RESults:PASSFail?

:POWer:HARMonics:RESults:RMS? → Query

Description	Returns the RMS value of the source.
-------------	--------------------------------------

Syntax	:POWer:HARMonics:RESults:RMS?
--------	-------------------------------

Return parameter	<NR3>	Unit = A
------------------	-------	----------

Example :POWer:HARMonics:RESults:RMS?

:POWer:HARMonics:RESults:THDF? → Query

Description	Returns the THDF as a percentage. THDF is the ratio of total harmonic distortion to the RMS value of the fundamental component of the source.
-------------	---

Syntax	:POWer:HARMonics:RESults:THDF?
--------	--------------------------------

Return parameter	<NR3>	%
------------------	-------	---

Example :POWer:HARMonics:RESults:THDF?

**:POWer:HARMonics:RESults:THDR?** → Query

**Description** Returns the THDF as a percentage. THDF is the ratio of total harmonic distortion to the RMS value of the source.

**Syntax** :POWer:HARMonics:RESults:THDR?

**Return parameter** <NR3> %

**Example** :POWer:HARMonics:RESults:THDR?

**:POWer:HARMonics:RESults:SAVe** Set →

**Description** Saves the harmonic results to USB. See the operation chapter for save details.

**Syntax** :POWer:HARMonics:RESults:SAVe

**:POWer:GATing** Set →  
→ Query

**Description** Sets the measurement gating area.

**Syntax** :POWer:GATing {OFF | SCREen | CURSor | ?}

<b>Parameter/ Return parameter</b>	OFF	Turn gating off.
	SCREen	Set the measurement gating to the screen width.
	CURSor	Set the measurement gating to between the cursors.

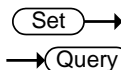
**Example** :POWer:GATing SCREen  
 Sets the measurement gating to the screen area.

**:POWer:QUALity:DISPlay** Set →  
→ Query

**Description** Sets the measurement display for the power quality measurements.

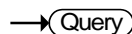


Syntax	:POWer:QUALity:DISPlay {OFF   NORMAl   INRUsh   BALLast   ALL   ?}	
Parameter/ Return parameter	OFF	Off
	NORMAl	Normal power quality measurements
	INRUsh	Inrush current related measurements
	BALLast	Ballast related measurements
	ALL	All power quality measurements
Example	:POWer:QUALity:DISPlay NORMAl Sets the measurement display to "normal".	



**:POWer:QUALity:FREQREference**

Description	Sets the frequency reference as the voltage or current source.	
Syntax	:POWer:QUALity:FREQREference {VOLTage   CURRent   ?}	
Parameter/ Return parameter	VOLTage	Voltage source.
	CURRent	Current source.
Example	:POWer:QUALity:FREQREference? >VOLTAGE	



**:POWer:QUALity:VMAX?**

Description	Returns the "VMAX".	
Syntax	:POWer:QUALity:VMAX?	
Return parameter	<NR3>	V
Example	:POWer:QUALity:VMAX? >1.5E+0	

**:POWER:QUALity:VMIN?**

→ Query

Description Returns the "VMIN".

Syntax :POWER:QUALity:VMIN?

Return parameter <NR3> V

Example :POWER:QUALity:VMIN?  
>0.5E-1

**:POWER:QUALity:IMAX?**

→ Query

Description Returns the "IMAX".

Syntax :POWER:QUALity:IMAX?

Return parameter <NR3> A

Example :POWER:QUALity:IMAX?  
>2.0E-2

**:POWER:QUALity:IMIN?**

→ Query

Description Returns the "IMIN".

Syntax :POWER:QUALity:IMIN?

Return parameter <NR3> A

Example :POWER:QUALity:IMIN?  
>0.5E-2

**:POWER:QUALity:DCVOLTage?**

→ Query

Description Returns the "DC Voltage".

Syntax :POWER:QUALity:DCVOLTage?

Return parameter <NR3> A

Example :POWer:QUALity:DCVOLTage?  
>1.11E-2

:POWer:QUALity:DCCURRent? → Query

Description Returns the “DC Current”.

Syntax :POWer:QUALity:DCCURRent?

Return parameter <NR3> A

Example :POWer:QUALity:DCCURRent?  
>1.5E-3

:POWer:QUALity:VCRESTfactor? → Query

Description Returns the “V Crest Factor”.

Syntax :POWer:QUALity:VCRESTfactor?

Return parameter <NR3>

Example :POWer:QUALity:VCRESTfactor?  
>1.41E+0

:POWer:QUALity:ICRESTfactor? → Query

Description Returns the “I Crest Factor”.

Syntax :POWer:QUALity:ICRESTfactor?

Return parameter <NR3>

Example :POWer:QUALity:ICRESTfactor?  
>3.06E+0

:POWer:QUALity:IMPedance? → Query

Description Returns the “Impedance”.

Syntax :POWer:QUALity:IMPedance?

Return parameter <NR3> Ω

Example :POWer:QUALity:IMPedance?

:POWer:QUALity:RESistance? → **Query**

Description Returns the “Resistance”.

Syntax :POWer:QUALity:RESistance?

Return parameter <NR3> Ω

Example :POWer:QUALity:RESistance?

:POWer:QUALity:REACtance? → **Query**

Description Returns the “Reactance”.

Syntax :POWer:QUALity:REACtance?

Return parameter <NR3> VAR

Example :POWer:QUALity:REACtance?

:POWer:QUALity:APPpwr? → **Query**

Description Returns the “Apparent Power”.

Syntax :POWer:QUALity:APPpwr?

Return parameter <NR3> Units = VA

Example :POWer:QUALity:APPpwr?  
>3.7E+1

:POWer:QUALity:FREQuency? → **Query**

Description Returns the “Frequency” of the input.

Syntax :POWer:QUALity:FREQuency?

Return parameter <NR3> Hz

Example :POWer:QUALity:FREQuency?  
>6.007E+1

:POWer:QUALity:ICREStfactor? → Query

Description Returns the current “Crest factor”.

Syntax :POWer:QUALity:ICREStfactor?

Return parameter <NR3>

Example :POWer:QUALity:ICREStfactor?  
>1.41E+0

:POWer:QUALity:IRMS? → Query

Description Returns the “I RMS”.

Syntax :POWer:QUALity:IRMS?

Return parameter <NR3> Unit=A

Example :POWer:QUALity:IRMS?  
>3.52E-2

:POWer:QUALity:PHASEangle? → Query

Description Returns the “Phase Angle”.

Syntax :POWer:QUALity:PHASEangle?

Return parameter <NR3> Unit=°

Example :POWer:QUALity:PHASEangle?  
>5.75E+1

:POWer:QUALity:POWERFActor? → Query

Description Returns the “Power Factor”.

Syntax :POWer:QUALity:POWERFActor?

Return parameter <NR3> 0~1

Example :POWer:QUALity:POWERFActor?  
>0.54E0

**:POWER:QUALity:REACTpwr?** → Query

Description	Returns the "Reactive Power".
Syntax	:POWER:QUALity:REACTpwr?
Return parameter	<NR3> Unit =VAR
Example	:POWER:QUALity:REACTpwr? >3.12E1


**:POWER:QUALity:TRUEpwr?** → Query

Description	Returns the "True Power".
Syntax	:POWER:QUALity:TRUEpwr?
Return parameter	<NR3> Unit =W
Example	:POWER:QUALity:TRUEpwr? >1.98E+1

**:POWER:QUALity:VRMS?** → Query

Description	Returns the "V RMS".
Syntax	:POWER:QUALity:VRMS?
Return parameter	<NR3> Unit =V
Example	:POWER:QUALity:VRMS? >1.04E+2

**:POWER:RIPple** Set →

Description	Performs a vertical autaset.
 Note	Only supported when the Ripple menu is turned on.
Syntax	:POWER:RIPple {VERTAUTOset}
Parameter	VERTAUTOset

Example :POWer:RIPple VERTAUTOset

:POWer:RIPple:RESults:AMPlitude? → Query

Description Returns the “Noise” amplitude.

Syntax :POWer:RIPple:RESults:AMPlitude?

Return parameter <NR3> Unit =A or V

Example :POWer:RIPple:RESults:AMPlitude?  
>1.15E+1

:POWer:RIPple:RESults:REALAmplitude? → Query

Description Returns the “Ripple” amplitude.

Syntax :POWer:RIPple:RESults:REALAmplitude?

Return parameter <NR3> Unit =A or V


Example :POWer:RIPple:RESults:REALAmplitude?  
>9.25E+1

Set →

:POWer:RIPple:SOURce → Query

Description Sets or queries the ripple source.

Syntax :POWer:RIPple:SOURce {VOLTage | CURRent | ?}

 Note Only supported when the Ripple menu is turned on.

Parameter/ VOLTage Voltage source

Return Parameter CURRent Current source

Example :POWer:RIPple:SOURce VOLTage

:POWer:INRUshcurrent:RESults:FIRStpeak? → Query

Description Returns the “First Peak” inrush current.

Syntax :POWer:INRUsh:RESults:FIRStpeak?

Return parameter <NR3> Unit =A

Example :POWer:INRUshcurrent:RESults:FIRStpeak?  
>2.44E+1

:POWer:INRUshcurrent:RESults:SECondpeak? → **Query**

Description Returns the “Second Peak” inrush current.

Syntax :POWer:INRUsh:RESults:SECondpeak?

Return parameter <NR3> Unit =A

Example :POWer:INRUsh:RESults:SECondpeak?  
>-2.36E+1

**Set** →

:POWer:MODulation:SOUrce

→ **Query**

Description Set or query the modulation source.

Syntax :POWer:MODulation:SOUrce {VOLTage|CURRent}  
:POWer:MODulation:SOUrce?

Parameter VOLTage Set modulation source as voltage.  
CURRent Set modulation source as current.

Example :POWer:MODulation:SOUrce Voltage  
:POWer:MODulation:SOUrce?

**Set** →

:POWer:MODulation:TYPE

→ **Query**

Description Set or query the modulation type.

Syntax :POWer:MODulation:TYPE {PWidth|NWidth|PERIod  
|PDUty|NDUty|FREQuency}POWer:MODulation:TYPE  
?

Parameter PWidth Set type as +width.  
NWidth Set type as -width.  
PERIod Set type as period.



PDuty	Set type as +duty.
NDuty	Set type as -duty.
FREQuency	Set type as frequency.

Example :POWer:MODulation:TYPe PWidth  
:POWer:MODulation:TYPe?

:POWer:REFLevel:PERCent:MID

Set →  
→ Query

Description Set or query the mid reference levels by percent.

Syntax :POWer:REFLevel:PERCent:MID <NRF>  
:POWer:REFLevel:PERCent:MID?

Parameter <NRF> 0-100%

Example :POWer:REFLevel:PERCent:MID 50  
:POWer:REFLevel:PERCent:MID?

Set →  
→ Query

Description Set or return the axes type.

Syntax :POWer:SOA:AXESTYPE {LOG|LINear}  
:POWer:SOA:AXESTYPE?

LOG	Set axes type as log.
LINear	Set axes type as linear.

Example :POWer:SOA:AXESTYPE LINear  
:POWer:SOA:AXESTYPE?

Set →  
→ Query

:POWer:SOA:LINear:XMAX

Description Sets or returns the linear max x axis.

Syntax :POWer:SOA:LINear:XMAX <NRF>  
:POWer:SOA:LINear:XMAX?

Parameter <NRF> -9e+3 ~ 10e+3. The value can't set lower than min x axis.

Example :POWer:SOA:LINear:XMAX 5.5E3  
:POWer:SOA:LINear:XMAX?

:POWer:SOA:LINear:XMIN

Set →

→ Query

Description Sets or returns the linear min x axis.

Syntax :POWer:SOA:LINear:XMIN <NRF>  
:POWer:SOA:LINear:XMIN?

Parameter <NRF> -10e+3 ~ 9e+3. The value can't set higher than max x axis.

Example :POWer:SOA:LINear:XMIN -2E-2  
:POWer:SOA:LINear:XMIN?

:POWer:SOA:LINear:YMAX

Set →

→ Query

Description Sets or returns the linear max y axis.

Syntax :POWer:SOA:LINear:YMAX <NRF>  
:POWer:SOA:LINear:YMAX?

Parameter <NRF> -9e+3 ~ 10e+3. The value can't set lower than min y axis.

Example :POWer:SOA:LINear:YMAX 8  
:POWer:SOA:LINear:YMAX?

:POWer:SOA:LINear:YMIN

Set →

→ Query

Description Sets or returns the linear min y axis.

Syntax :POWer:SOA:LINear:YMIN <NRF>  
:POWer:SOA:LINear:YMIN?

Parameter <NRF> -10e+3 ~ 9e+3. The value can't set higher than max y axis.

Example :POWer:SOA:LINear:YMIN 0.5  
:POWer:SOA:LINear:YMIN?

Set →

→ Query

:POWer:SOA:LOG:XMAX

Description Sets or returns the log max x axis.

Syntax :POWer:SOA:LOG:XMAX <NRF>  
:POWer:SOA:LOG:XMAX?

Parameter <NRF> 0.001, 0.01, 0.1, 1, 10, 100, 1000, 10000

Example :POWer:SOA:LOG:XMAX 1000  
:POWer:SOA:LOG:XMAX?

Set →

→ Query

:POWer:SOA:LOG:XMIN

Description Sets or returns the log min x axis.

Syntax :POWer:SOA:LOG:XMIN <NRF>  
:POWer:SOA:LOG:XMIN?

Parameter <NRF> 0.001, 0.01, 0.1, 1,10, 100, 1000, 10000

Example :POWer:SOA:LOG:XMIN 0.1  
:POWer:SOA:LOG:XMIN?

Set →

→ Query

:POWer:SOA:LOG:YMAX

Description Sets or returns the log max y axis.

Syntax :POWer:SOA:LOG:YMAX <NRF>  
:POWer:SOA:LOG:YMAX?

Parameter <NRF> 0.001, 0.01, 0.1, 1,10, 100, 1000, 10000

Example :POWer:SOA:LOG:YMAX 100  
:POWer:SOA:LOG:YMAX?

Set →  
 → Query

**:POWer:SOA:LOG:YMIN**

---

Description	Sets or returns the log min y axis.
Syntax	:POWer:SOA:LOG:YMIN <NRF> :POWer:SOA:LOG:YMIN?
Parameter	<NRF> 0.001, 0.01, 0.1, 1,10, 100, 1000, 10000
Example	:POWer:SOA:LOG:YMIN 0.001 :POWer:SOA:LOG:YMIN?

Set →  
 → Query

**:POWer:SOA:MASK:TYPe**

---

Description	Sets or returns the mask type in SOA.
Syntax	:POWer:SOA:MASK:TYPe {LIMits POINt} :POWer:SOA:MASK:TYPe?
Parameter	LIMits Set mask type as limits. POINt Set mask type as points.
Example	:POWer:SOA:MASK:TYPe LIMits :POWer:SOA:MASK:TYPe?

Set →  
 → Query

**:POWer:SOA:MASK:NUMPt**

---

Description	Sets or returns the number of points for the SOA mask.
Syntax	:POWer:SOA:MASK:NUMPt <NR> :POWer:SOA:MASK:NUMPt?
Parameter	<NR> 2~10
Example	:POWer:SOA:MASK:NUMPt 6 :POWer:SOA:MASK:NUMPt?

 →  
 → 

**:POWer:SOA:MASK:POINT<x>**

Description	Sets or returns the point's coordinate for the mask.	
Syntax	:POWer:SOA:MASK:POINT<x> {<XMASK> <YMASK>} :POWer:SOA:MASK:POINT<x>?	
Parameter	<XMASK>	Set x coordinate of the point.
	<YMASK>	Set y coordinate of the point.Range: -10k ~ 10k
	<x>	Point 1~10
Example	:POWer:SOA:MASK:POINT2 50,30 :POWer:SOA:MASK:POINT2?	

 →  
 → 

**:POWer:SOA:MASK:MAXVoltage**

Description	Sets or returns the max voltage for mask's limit.	
Syntax	:POWer:SOA:MASK:MAXVoltage <NRF> :POWer:SOA:MASK:MAXVoltage?	
Parameter	<NRF>	1m~10k
Example	:POWer:SOA:MASK:MAXVoltage 100 :POWer:SOA:MASK:MAXVoltage?	

 →  
 → 

**:POWer:SOA:MASK:MAXCURREnt**

Description	Sets or returns the max current for mask's limit.	
Syntax	:POWer:SOA:MASK:MAXCURREnt <NRF> :POWer:SOA:MASK:MAXCURREnt?	
Parameter	<NRF>	1m~10k
Example	:POWer:SOA:MASK:MAXCURREnt 100 :POWer:SOA:MASK:MAXCURREnt?	

:POWer:SOA:MASK:MAXPower (Set) →  
→ (Query)

Description	Sets or returns the max power for mask's limit.	
Syntax	:POWer:SOA:MASK:MAXPower <NRF> :POWer:SOA:MASK:MAXPower?	
Parameter	<NRF>	1m~10k. The value can't set higher than (max voltage)*(max current).
Example	:POWer:SOA:MASK:MAXPower 50 :POWer:SOA:MASK:MAXPower?	

:POWer:SOA:MASK:STOPOnviol (Set) →  
→ (Query)

Description	Sets or returns the state of stop on violation.	
Syntax	:POWer:SOA:MASK:STOPOnviol {OFF ON} :POWer:SOA:MASK:STOPOnviol?	
Parameter	ON	Stop on violation.
	OFF	Continue on violation.
Example	:POWer:SOA:MASK:STOPOnviol ON :POWer:SOA:MASK:STOPOnviol?	

:POWer:SOA:RESults:FAILures? → (Query)

Description	Returns the values of waveforms acquired.	
Syntax	:POWer:SOA:RESults:FAILures?	
Example	:POWer:SOA:RESults:FAILures?	

:POWer:SOA:RESults:NUMACq? → (Query)

Description	Returns the values of failing samples.	
Syntax	:POWer:SOA:RESults:NUMACq?	
Example	:POWer:SOA:RESults: NUMACq?	

**:POWer:SOA:MASK:RESults:STATe?** → **Query**

Description	Returns the mask test state.
Syntax	:POWer:SOA:MASK:RESults:STATe?
Example	:POWer:SOA:MASK:RESults:STATe?

**Set** →

**:POWer:SWLoss:CONDCALCtype** → **Query**

Description	Sets or returns the type of conduction calculation.	
Syntax	:POWer:SWLoss:CONDCALCtype {VOLTage RDSon VCEsat}	
	:POWer:SWLoss:CONDCALCtype?	
Parameter	VOLTage	Set the type as voltage waveform.
	RDSon	Set the type as RDS(sat)(Best for FJT/IGBT).
	VCEsat	Set the type as VCE(sat)
Example	:POWer:SWLoss:CONDCALCtype RDSon	
	:POWer:SWLoss:CONDCALCtype?	

**Set** →

**:POWer:SWLoss:DISPlay** → **Query**

Description	Sets or returns the type of switching loss measure display.	
Syntax	:POWer:SWLoss:DISPlay {ALL ENERgy POWER}	
	:POWer:SWLoss:DISPlay?	
Parameter	ALL	Display all results.
	ENERgy	Display results of energy loss.
	POWER	Display results of power loss.
Example	:POWer:SWLoss:DISPlay POWER	
	:POWer:SWLoss:DISPlay?	

Set →  
 → Query

**:POWer:SWLoss:GATing**

Description	Sets or returns the swiching loss gating.	
Syntax	:POWer:SWLoss:GATing {OFF SCREEn CURSor} :POWer:SWLoss:GATing?	
Parameter	OFF	Full record
	SCREEn	Gating set to screen width
	CURSor	Gating between cursors
Example	:POWer:SWLoss:GATing SCREEn :POWer:SWLoss:GATing?	

Set →  
 → Query

**:POWer:SWLoss:RDSON**

Description	Sets or returns the value of RDS(on).	
Syntax	:POWer:SWLoss:RDSON <NRF> :POWer:SWLoss:RDSON?	
Parameter	<NRF>	Sets the value of RDS(on). Range:0 ~ 100 ohm.
Example	:POWer:SWLoss:RDSON 0.25 :POWer:SWLoss:RDSON?	

Set →  
 → Query

**:POWer:SWLoss:VCEsat**

Description	Sets or returns the value of VCE(sat).	
Syntax	:POWer:SWLoss:VCEsat <NRF> :POWer:SWLoss:VCEsat?	
Parameter	<NRF>	Sets the value of VCE(sat). Range:0 ~100 V.
Example	:POWer:SWLoss:VCEsat 0.56 :POWer:SWLoss:VCEsat?	



**:POWer:SWLoss:REFLevel:PERCent:MID** 


**Description** Sets or returns the mid reference levels by percent.

**Syntax** :POWer:SWLoss:REFLevel:PERCent:Mid <NRF>  
:POWer:SWLoss:REFLevel:PERCent:Mid?

**Parameter** <NRF> 0 ~100%

**Example** :POWer:SWLoss:REFLevel:PERCent:Mid 40  
:POWer:SWLoss:REFLevel:PERCent:Mid?

**:POWer:SWLoss:REFLevel:PERCent:CURRent** 


**Description** Sets or returns the current levels by percent.

**Syntax** :POWer:SWLoss:REFLevel:PERCent:CURRent <NRF>  
:POWer:SWLoss:REFLevel:PERCent:CURRent?

**Parameter** <NRF> 0 ~100%

**Example** :POWer:SWLoss:REFLevel:PERCent:CURRent 35.5  
:POWer:SWLoss:REFLevel:PERCent:CURRent

**:POWer:SWLoss:REFLevel:PERCent:VOLTage** 


**Description** Sets or returns the voltage levels by percent.

**Syntax** :POWer:SWLoss:REFLevel:PERCent:VOLTage <NRF>  
:POWer:SWLoss:REFLevel:PERCent:VOLTage?

**Parameter** <NRF> 0 ~100%

**Example** :POWer:SWLoss:REFLevel:PERCent:VOLTage 65.5  
:POWer:SWLoss:REFLevel:PERCent:VOLTage

**:POWer:SWLoss:CONDuction:ENERgy:MAX?** 


**Description** Returns the max conduction of energy loss.

Syntax :POWER:SWLoss:CONDUCTION:ENERgy:MAX?

Example :POWER:SWLoss:CONDUCTION:ENERgy:MAX?

:POWER:SWLoss:CONDUCTION:ENERgy:MEAN? → **Query**

Description Returns the mean conduction of energy loss.

Syntax :POWER:SWLoss:CONDUCTION:ENERgy:MEAN?

Example :POWER:SWLoss:CONDUCTION:ENERgy:MEAN?

:POWER:SWLoss:CONDUCTION:ENERgy:MIN? → **Query**

Description Returns the min. conduction of energy loss.

Syntax :POWER:SWLoss:CONDUCTION:ENERgy:MIN?

Example :POWER:SWLoss:CONDUCTION:ENERgy:MIN?

:POWER:SWLoss:CONDUCTION:POWER:MAX? → **Query**

Description Returns the max conduction of power loss.

Syntax :POWER:SWLoss:CONDUCTION:POWER:MAX?

Example :POWER:SWLoss:CONDUCTION:POWER:MAX?

:POWER:SWLoss:CONDUCTION:POWER:MEAN? → **Query**

Description Returns the mean conduction of power loss.

Syntax :POWER:SWLoss:CONDUCTION:POWER:MEAN?

Example :POWER:SWLoss:CONDUCTION:POWER:MEAN?

:POWER:SWLoss:CONDUCTION:POWER:MIN? → **Query**

Description Returns the min. conduction of power loss.

Syntax :POWER:SWLoss:CONDUCTION:POWER:MIN?

Example :POWER:SWLoss:CONDUCTION:POWER:MIN?

:POWer:SWLoss:TOFF:ENERgy:MAX? → Query

---

Description Returns the max Toff of energy loss.

---

Syntax :POWer:SWLoss:TOFF:ENERgy:MAX?

---

Example :POWer:SWLoss:TOFF:ENERgy:MAX?

:POWer:SWLoss:TOFF:ENERgy:MEAN? → Query

---

Description Returns the mean Toff of energy loss.

---

Syntax :POWer:SWLoss:TOFF:ENERgy:MEAN?

---

Example :POWer:SWLoss:TOFF:ENERgy:MEAN?

:POWer:SWLoss:TOFF:ENERgy:MIN? → Query

---

Description Returns the min. Toff of energy loss.

---

Syntax :POWer:SWLoss:TOFF:ENERgy:MIN?

---

Example :POWer:SWLoss:TOFF:ENERgy:MIN?

:POWer:SWLoss:TOFF:POWer:MAX? → Query

---

Description Returns the max Toff of power loss.

---

Syntax :POWer:SWLoss:TOFF:POWer:MAX?

---

Example :POWer:SWLoss:TOFF:POWer:MAX?

:POWer:SWLoss:TOFF:POWer:MEAN? → Query

---

Description Returns the mean Toff of power loss.

---

Syntax :POWer:SWLoss:TOFF:POWer:MEAN?

---

Example :POWer:SWLoss:TOFF:POWer:MEAN?

:POWer:SWLoss:TOFF:POWer:MIN? → Query

Description Returns the min. Toff of power loss.

Syntax :POWer:SWLoss:TOFF:POWer:MIN?

Example :POWer:SWLoss:TOFF:POWer:MIN?

:POWer:SWLoss:TON:ENERgy:MAX? → Query

Description Returns the max Ton of energy loss.

Syntax :POWer:SWLoss:TON:ENERgy:MAX?

Example :POWer:SWLoss:TON:ENERgy:MAX?

:POWer:SWLoss:TON:ENERgy:MEAN? → Query

Description Returns the mean Ton of energy loss.

Syntax :POWer:SWLoss:TON:ENERgy:MEAN?

Example :POWer:SWLoss:TON:ENERgy:MEAN?

:POWer:SWLoss:TON:ENERgy:MIN? → Query

Description Returns the min. Ton of energy loss.

Syntax :POWer:SWLoss:TON:ENERgy:MIN?

Example :POWer:SWLoss:TON:ENERgy:MIN?

:POWer:SWLoss:TON:POWer:MAX? → Query

Description Returns the max Ton of power loss.

Syntax :POWer:SWLoss:TON:POWer:MAX?

Example :POWer:SWLoss:TON:POWer:MAX?

:POWer:SWLoss:TON:POWer:MEAN? → Query

Description Returns the mean Ton of power loss.

Syntax :POWer:SWLoss:TON:POWer:MEAN?

Example :POWer:SWLoss:TON:POWer:MEAN?

:POWer:SWLoss:TON:POWer:MIN? → Query

Description Returns the min. Ton of power loss.

Syntax :POWer:SWLoss:TON:POWer:MIN?

Example :POWer:SWLoss:TON:POWer:MIN?

:POWer:SWLoss:TOTAl:ENERgy:MAX? → Query

Description Returns the max. total value of energy loss.

Syntax :POWer:SWLoss:TOTAl:ENERgy:MAX?

Example :POWer:SWLoss:TOTAl:ENERgy:MAX?

:POWer:SWLoss:TOTAl:ENERgy:MEAN? → Query

Description Returns the mean total value of energy loss.

Syntax :POWer:SWLoss:TOTAl:ENERgy:MEAN?

Example :POWer:SWLoss:TOTAl:ENERgy:MEAN?

:POWer:SWLoss:TOTAl:ENERgy:MIN? → Query

Description Returns the min. total value of energy loss.

Syntax :POWer:SWLoss:TOTAl:ENERgy:MIN?

Example :POWer:SWLoss:TOTAl:ENERgy:MIN?

**:POWer:SWLoss:TOTal:POWer:MAX?** → **Query**

Description	Returns the max total value of power loss.
Syntax	:POWer:SWLoss:TOTal:POWer:MAX?
Example	:POWer:SWLoss:TOTal:POWer:MAX?

**:POWer:SWLoss:TOTal:POWer:MEAN?** → **Query**

Description	Returns the mean total value of power loss.
Syntax	:POWer:SWLoss:TOTal:POWer:MEAN?
Example	:POWer:SWLoss:TOTal:POWer:MEAN?

**:POWer:SWLoss:TOTal:POWer:MIN?** → **Query**

Description	Returns the min. total value of power loss.
Syntax	:POWer:SWLoss:TOTal:POWer:MIN?
Example	:POWer:SWLoss:TOTal:POWer:MIN?

**:POWer:TRANSient:DURation** → **Query**

**Set** →

Description	Sets or returns the value duration.
Syntax	:POWer:TRANSient:DURation <NRF> :POWer:TRANSient:DURation?
Parameter	<NRF> 10n ~ 1000
Example	:POWer:TRANSient:DURation 2e-3 :POWer:TRANSient:DURation?

**Set** →

**:POWer:TRANSient:OVERShoot** → **Query**

Description	Sets or returns the value of overshoot.
-------------	---

Syntax :POWer:TRANSient:OVERShoot <NRF>  
:POWer:TRANSient:OVERShoot?

Parameter <NRF> 0.1 ~ 100(%)

Example :POWer:TRANSient:OVERShoot 25.5  
:POWer:TRANSient:OVERShoot?

Set →

→ Query

:POWer:TRANSient:STEADYVout

Description Sets or returns the value of steady Vout.

Syntax :POWer:TRANSient:STEADYVout <NRF>  
:POWer:TRANSient:STEADYVout?

Parameter <NRF> 0.01 ~ 100(V)

Example :POWer:TRANSient:STEADYVout 2.2  
:POWer:TRANSient:STEADYVout?

Set →

→ Query

:POWer:TRANSient:LOWCurrent

Description Sets or returns the value of low current.

Syntax :POWer:TRANSient:LOWCurrent <NRF>  
:POWer:TRANSient:LOWCurrent?

Parameter <NRF> 0.01 ~ 100(A)

Example :POWer:TRANSient:LOWCurrent 0.06  
:POWer:TRANSient:LOWCurrent?

Set →

→ Query

:POWer:TRANSient:HIGHCurrent

Description Sets or returns the value of high current.

Syntax :POWer:TRANSient:HIGHCurrent <NRF>  
:POWer:TRANSient:HIGHCurrent?

Parameter <NRF> 0.01 ~ 100(A)

Example :POWer:TRANSient:HIGHCurrent 0.08  
 :POWer:TRANSient:HIGHCurrent?

:POWer:TRANSient:RESults:VALue? → Query

Description Returns the result of transient test.

Syntax :POWer:TRANSient:RESults:VALue?

Example :POWer:TRANSient:RESults:VALue?

Set →

:POWer:EFFiciency:STATIstics:MODE → Query

Description Sets or returns the state of efficiency statistics.

Syntax :POWer:EFFiciency:STATIstics:MODE {OFF|ON}  
 :POWer:EFFiciency:STATIstics:MODE?

Parameter	OFF	Turn off the statistics.
	ON	Turn on the statistics.

Example :POWer:EFFiciency:STATIstics:MODE ON  
 :POWer:EFFiciency:STATIstics:MODE?

Set →

:POWer:EFFiciency:STATIstics:WEIghting → Query

Description Sets or returns the weighting(mean & std dev samples) of efficiency statistics.

Syntax :POWer:EFFiciency:STATIstics:WEIghting <NR1>  
 :POWer:EFFiciency:STATIstics:WEIghting?

Parameter <NR1> 2 ~ 1000

Example :POWer:EFFiciency:STATIstics:WEIghting 625  
 :POWer:EFFiciency:STATIstics:WEIghting?

:POWer:EFFiciency:STATIstics Set →

Description Resets the efficiency statistics.



Syntax	:POWer:EFFiciency:STATIstics {RESET}
Parameter	RESET    Reset the statistics.
Example	:POWer:EFFiciency:STATIstics RESET

**:POWer:EFFiciency:INPUTPOWer:VALue?** → Query

Description	Returns the value of input power.
Syntax	:POWer:EFFiciency:INPUTPOWer:VALue?
Example	:POWer:EFFiciency:INPUTPOWer:VALue?

**:POWer:EFFiciency:INPUTPOWer:MEAN?** → Query

Description	Returns the mean value of input power.
Syntax	:POWer:EFFiciency:INPUTPOWer:MEAN?
Example	:POWer:EFFiciency:INPUTPOWer:MEAN?

**:POWer:EFFiciency:INPUTPOWer:MINImum?** → Query

Description	Returns the minimum value of input power.
Syntax	:POWer:EFFiciency:INPUTPOWer:MINImum?
Example	:POWer:EFFiciency:INPUTPOWer:MINImum?

**:POWer:EFFiciency:INPUTPOWer:MAXmum?** → Query

Description	Returns the maximum value of input power.
Syntax	:POWer:EFFiciency:INPUTPOWer:MAXimum?
Example	:POWer:EFFiciency:INPUTPOWer:MAXimum?

**:POWer:EFFiciency:INPUTPOWer:STDdev?** → Query

Description	Returns the std dev value of input power.
Syntax	:POWer:EFFiciency:INPUTPOWer:STDdev?

Example :POWer:EFFiciency:INPutPOWer:STDeV?

:POWer:EFFiciency:OUTPutPOWer:VALue? → **Query**

Description Returns the value of output power.

Syntax :POWer:EFFiciency:OUTPutPOWer:VALue?

Example :POWer:EFFiciency:OUTPutPOWer:VALue?

:POWer:EFFiciency:OUTPutPOWer:MEAN? → **Query**

Description Returns the mean value of output power.

Syntax :POWer:EFFiciency:OUTPutPOWer:MEAN?

Example :POWer:EFFiciency:OUTPutPOWer:MEAN?

:POWer:EFFiciency:OUTPutPOWer:MINImum? → **Query**

Description Returns the minimum value of output power.

Syntax :POWer:EFFiciency:OUTPutPOWer:MINImum?

Example :POWer:EFFiciency:OUTPutPOWer:MINImum?

:POWer:EFFiciency:OUTPutPOWer:MAXmum? → **Query**

Description Returns the maximum value of output power.

Syntax :POWer:EFFiciency:OUTPutPOWer:MAXimum?

Example :POWer:EFFiciency:OUTPutPOWer:MAXimum?

:POWer:EFFiciency:OUTPutPOWer:STDeV? → **Query**

Description Returns the std dev value of output power.

Syntax :POWer:EFFiciency:OUTPutPOWer:STDeV?

Example :POWer:EFFiciency:OUTPutPOWer:STDeV?

**:POWer:EFFiciency:RESults:VALue?** → Query

---

Description Returns the value of efficiency test results.

---

Syntax :POWer:EFFiciency:RESults:VALue?

---

Example :POWer:EFFiciency:RESults:VALue?

**:POWer:EFFiciency:RESults:MEAN?** → Query

---

Description Returns the mean value of efficiency test results.

---

Syntax :POWer:EFFiciency:RESults:MEAN?

---

Example :POWer:EFFiciency:RESults:MEAN?

**:POWer:EFFiciency:RESults:MINImum?** → Query

---

Description Returns the minimum value of efficiency test results.

---

Syntax :POWer:EFFiciency:RESults:MINImum?

---

Example :POWer:EFFiciency:RESults:MINImum?

**:POWer:EFFiciency:RESults:MAXimum?** → Query

---

Description Returns the maximum value of efficiency test results.

---

Syntax :POWer:EFFiciency:RESults:MAXimum?

---

Example :POWer:EFFiciency:RESults:MAXimum?

**:POWer:EFFiciency:RESults:STDdev?** → Query

---

Description Returns the std dev value of efficiency test results.

---

Syntax :POWer:EFFiciency:RESults:STDdev?

---

Example :POWer:EFFiciency:RESults:STDdev?

:POWer:BHCurve:WINDings (Set) →  
→ (Query)

Description	Set or query the value of windings
Syntax	:POWer:BHCurve:WINDings <NR1> :POWer:BHCurve:WINDings?
Parameter	<NR1> Value of windings. Range:1 ~ 1M
Example	:POWer:BHCurve:WINDings 80 :POWer:BHCurve:WINDings?

:POWer:BHCurve:CROSECArea (Set) →  
→ (Query)

Description	Set or query the value of cross section area.
Syntax	:POWer:BHCurve:CROSECArea <NRF> :POWer:BHCurve:CROSECArea?
Parameter	<NRF> Value of cross section area. Range:1n ~ 1M.
Example	:POWer:BHCurve:CROSECArea 0.35 :POWer:BHCurve:CROSECArea?

:POWer:BHCurve:MAGNLength (Set) →  
→ (Query)

Description	Set or query the value of magnetic length.
Syntax	:POWer:BHCurve:MAGNLength <NRF> :POWer:BHCurve:MAGNLength?
Parameter	<NRF> Set the magnetic length. Range:1m ~ 100
Example	:POWer:BHCurve:MAGNLength 20.5 :POWer:BHCurve:MAGNLength?

 →  
 → 


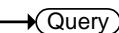
**:POWer:BHCurve:B:POStion**

Description	Set or query the B curve position.
Syntax	:POWer:BHCurve:B:POStion <NRF> :POWer:BHCurve:B:POStion?
Parameter	<NRF> Set the B curve position. Range:-12.0 ~ 12.0
Example	:POWer:BHCurve:B:POStion 6 :POWer:BHCurve:B:POStion?

 →  
 → 

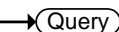
**:POWer:BHCurve:B:SCAlE**

Description	Set or query the B curve scale.
Syntax	:POWer:BHCurve:B:SCAlE <NRF> :POWer:BHCurve:B:SCAlE?
Parameter	<NRF> Set the B curve scale.
Example	:POWer:BHCurve:B:SCAlE 2 :POWer:BHCurve:B:SCAlE?

 →  
 → 

**:POWer:BHCurve:H:POStion**

Description	Set or query the H curve position.
Syntax	:POWer:BHCurve:H:POStion <NRF> :POWer:BHCurve:H:POStion?
Parameter	<NRF> Set or query the H curve position
Example	:POWer:BHCurve:H:POStion 20.5 :POWer:BHCurve:H:POStion?

 →  
 → 

**:POWer:BHCurve:H:SCAlE**

Description	Set or query the H curve scale.
-------------	---------------------------------

Syntax :POWer:BHCurve:H:SCALE <NRF>  
:POWer:BHCurve:H:SCALE?

Parameter <NRF> Set the H curve scale.

Example :POWer:BHCurve:H:SCALE 10  
:POWer:BHCurve:H:SCALE?

Set →

→ Query

:POWer:TONOff:TEST

Description Sets or returns the type of turn on/off test.

Syntax :POWer:TONOff:TEST {TOFF|TON}  
:POWer:TONOff:TEST?

Parameter TOFF Set the test off.  
TON Set the test on.

Example :POWer:TONOff:TEST TON  
:POWer:TONOff:TEST?

Set →

→ Query

:POWer:TONOff:SETup:DURation

Description Sets or returns the duration of turn on/off test.

Syntax :POWer:TONOff:SETup:DURation{USER|2|1|0.5}  
:POWer:TONOff:SETup:DURation?

Parameter USER Set duration to user define value.  
2 Set duration to 2sec.  
1 Set duration to 1sec.  
0.5 Set duration to 0.5sec

Example :POWer:TONOff:SETup:DURation 2  
:POWer:TONOff:SETup:DURation?

Set →

:POWer:TONOff:SETup:APply

Description Apply the turn on/off test setup.

Syntax :POWer:TONOff:SETup:APPLY

Example :POWer:TONOff:SETup:APPLY

**:POWer:TONOff:SETup:SAVESetup**

Set →

Description Save the turn on/off test setup.

Syntax :POWer:TONOff:SETup:SAVESetup

Example :POWer:TONOff:SETup:SAVESetup

**:POWer:TONOff:DISPlay**

Set →

→ Query

Description Sets or returns the type of turn on/off measure display.

Syntax :POWer:TONOff:DISPlay {OFF|TDElay|RTIME|?}  
:POWer:TONOff:DISPlay?

Parameter	OFF	Turn off measure display.
	TDElay	Display turn on/off delay measure.
	RTIME	Display rise time measure.

Example :POWer:TONOff:DISPlay 2  
:POWer:TONOff:DISPlay?

Set →

→ Query

**:POWer:SHOrt:DISPlay**

Description Sets or returns the state of short test display.

Syntax :POWer:SHOrt:DISPlay {OFF|ON}  
:POWer:SHOrt:DISPlay?

Parameter	OFF	Set the display off.
	ON	Set the display on.

Example :POWer:SHOrt:DISPlay ON  
:POWer:SHOrt:DISPlay?

**:POWer:SHORt:APPLY**

Set →

Description      Apply the setup of short test.

Syntax             :POWer:SHORt:APPLY

Example            :POWer:SHORt:APPLY

**:POWer:OCP:DISPlay**

Set →

→ Query

Description      Sets or returns the state of OCP test display.

Syntax             :POWer:OCP:DISPlay {OFF|ON}  
                       :POWer:OCP:DISPlay?

Parameter	OFF	Set the display off.
	ON	Set the display on.

Example            :POWer:OCP:DISPlay ON  
                       :POWer:OCP:DISPlay?

**:POWer:OCP:APPLY**

Set →

Description      Apply the setup of OCP test.

Syntax             :POWer:OCP:APPLY

Example            :POWer:OCP:APPLY

**:POWer:PSRR:RUN**

Set →

→ Query

Description      Runs the PSRR function or returns the PSRR state.

Syntax             :POWer:PSRR:RUN  
                       :POWer:PSRR:RUN?

Example            :POWer:PSRR:RUN



**:POWer:PSRR:STOP** 


Description	Stops the PSRR function or returns the PSRR state.
Syntax	:POWer:PSRR:STOP :POWer:PSRR:STOP?
Example	:POWer:PSRR:STOP

**:POWer:PSRR:SOURce:INPut** 


Description	Sets or returns the input source for PSRR.
Syntax	:POWer:PSRR:SOURce:INPut {CH1~CH4} :POWer:PSRR:SOURce:INPut?
Parameter	<b>CH1~CH4</b> Channel 1 to Channel 4
Example	:POWer:PSRR:SOURce:INPut CH1 Set the input source as channel 1.

**:POWer:PSRR:SOURce:OUTPut** 


Description	Sets or returns the output source for PSRR.
Syntax	:POWer:PSRR:SOURce:OUTPut {CH1~CH4} :POWer:PSRR:SOURce:OUTPut?
Parameter	<b>CH1~CH4</b> Channel 1 to Channel 4
Example	:POWer:PSRR:SOURce:OUTPut CH2 Set the output source as channel 2.

**:POWer:PSRR:FREQuency:STARt** 


Description	Sets or returns the start frequency for PSRR.
Syntax	:POWer:PSRR:FREQuency:STARt {<NRF>} :POWer:PSRR:FREQuency:STARt?

Parameter <NRf> Sets the frequency to use.  
(Range:20Hz~25MHz)

Example :POWer:PSRR:FREQuency:STARt 100  
Sets the start frequency as 100Hz.

→  
 →

:POWer:PSRR:FREQuency:STOP

Description Sets or returns the stop frequency for PSRR.

Syntax :POWer:PSRR:FREQuency:STOP {<NRf>}  
:POWer:PSRR:FREQuency:STOP?

Parameter <NRf> Sets the frequency to use.  
(Range:20Hz~25MHz)

Example :POWer:PSRR:FREQuency:STOP 500  
Sets the stop frequency as 500Hz.

→  
 →

:POWer:PSRR:AWG:LOAD

Description Sets or returns the impedance for load.

Syntax :POWer:PSRR:AWG:LOAD {FIFTy|HIGHZ}  
:POWer:PSRR:AWG:LOAD?

Parameter FIFTy 50 ohm  
HIGHZ High impedance

Example :POWer:PSRR:AWG:LOAD HIGHZ  
Sets the load as high impedance.

→  
 →

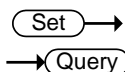
:POWer:PSRR:AWG:AMPlitude

Description Sets or returns the amplitude for PSRR. When amplitude profile is on, sets or returns the amplitude for the selected frequency.

Syntax :POWer:PSRR:AWG:AMPlitude {<NRf>,<range>}  
:POWer:PSRR:AWG:AMPlitude? [<range>]

Parameter	<p>&lt;NRf&gt; Sets the amplitude to use.(Load=FIFTY, Range:0.01Vpp~2.5Vpp; Load=HIGHZ, Range:0.02Vpp~5Vpp)</p> <p>&lt;range&gt; The selected frequency {F20hz   F100hz   F1Khz   F10Khz   F100Khz   F1Mhz   F10Mhz   F25Mhz}</p> <p>F20hz Frequency range &gt;20Hz (The default &lt;range&gt;)  The selected frequency. {F20hz   F100hz   F1Khz   F10Khz   F100Khz   F1Mhz   F10Mhz   F25Mhz}  F20hz: Frequency range &gt;20Hz (The default &lt;range&gt;).  F100hz: Frequency range &gt;100Hz. F1Khz: Frequency range &gt;1kHz. F10Khz: Frequency range &gt;10kHz. F100Khz: Frequency range &gt;100kHz. F1Mhz: Frequency range &gt;1MHz. F10Mhz: Frequency range &gt;10MHz. F25Mhz: Frequency range 25MHz.</p>
-----------	--

Example :POWer:PSRR:AWG:AMPlitude 0.2  
Sets the amplitude as 0.2Vpp.  
:POWer:PSRR:AWG:AMPlitude 0.5,F100HZ  
:POWer:PSRR:AWG:AMPlitude? F100HZ  
0.5



:POWer:PSRR:POINT

Description Sets or returns the number of processing points in a decade.

Syntax :POWer:PSRR:POINT {<NR1>}  
:POWer:PSRR:POINT?

Parameter	<code>&lt;NR1&gt;</code>	The number of points in a decade. (Range:10, 15, 30, 45, 90)
-----------	--------------------------	--

Example	:POWer:PSRR:POINt 15	Sets the number of processing points as 15 in a decade.
---------	----------------------	---

**:POWer:PSRR:SAVe** (Set) →

Description	Saves the PSRR result.
-------------	------------------------

Syntax	:POWer:PSRR:SAVe
--------	------------------

Example	:POWer:PSRR:SAVe	Saves the result to default file.
---------	------------------	-----------------------------------

**:POWer:PSRR:RECALL** (Set) →

Description	Recalls the PSRR result from memory or USB.
-------------	---

Syntax	:POWer:PSRR:RECALL {<file path> ("Disk:/xxx.FRD", "USB:/xxx.FRD")}
--------	---

Parameter	<code>xxx.FRD</code>	Filename
-----------	----------------------	----------

Example	:POWer:PSRR:RECALL "Disk:/PSR0001.FRD"	Recalls a PSRR result named PSR0001.FRD from root directory (Disk:/) of the scope.
---------	--	--

**:POWer:PSRR:DATA** → (Query)

Description	Shows the detailed information of PSRR settings and results.
-------------	--

Syntax	:POWer:PSRR:DATA?
--------	-------------------

Example	:POWer:PSRR:DATA?	Shows the PSRR result's detail.
---------	-------------------	---------------------------------

**:POWer:PSRR:SAVETOCsv** (Set) →

Description	Saves the PSRR result as a CSV file.
Syntax	:POWer:PSRR:SAVETOCsv
Example	:POWer:PSRR:SAVETOCsv Saves results as CSV file.

**:POWer:PSRR:AWG:AMPlitude:PROFile** (Set) →  
→ (Query)

Description	Sets or returns the state of amplitude profile.
Syntax	:POWer:PSRR:AWG:AMPlitude:PROFile {ON OFF} :POWer:PSRR:AWG:AMPlitude:PROFile?
Parameter	ON Turn on the profile. OFF Turn off the profile.
Example	:POWer:PSRR:AWG:AMPlitude:PROFile ON :POWer:PSRR:AWG:AMPlitude:PROFile? ON

**:POWer:PSRR:AWG:AMPlitude:INTERPolation** (Set) →  
→ (Query)

Description	Sets or returns the state of linear interpolation for the selected frequency.
Syntax	:POWer:PSRR:AWG:AMPlitude:INTERPolation {ON OFF,[<range>]} :POWer:PSRR:AWG:AMPlitude:INTERPolation? [<range>]
Parameter	ON Turn on the interpolation. OFF Turn off the interpolation. <range> The selected frequency {F20hz   F100hz   F1Khz   F10Khz   F100Khz   F1Mhz   F10Mhz   F25Mhz}

F20hz: Frequency range >20Hz (The default <range>).  
 F100hz: Frequency range >100Hz.  
 F1Khz: Frequency range >1kHz.  
 F10Khz: Frequency range >10kHz.  
 F100Khz: Frequency range >100kHz.  
 F1Mhz: Frequency range >1MHz.  
 F10Mhz: Frequency range >10MHz.  
 F25Mhz: Frequency range 25MHz.

Example :POWer:PSRR:AWG:AMPlitude:INTERPolation  
 ON,F100HZ  
 :POWer:PSRR:AWG:AMPlitude:INTERPolation?  
 F100HZ  
 ON

:POWer:PSRR:DATA:GMARgin → Query

Description Returns the gain margin of PSRR results.

Syntax :POWer:PSRR:DATA:GMARgin?

Example :POWer:PSRR:DATA:GMARgin?

:POWer:PSRR:DATA:GMARgin:FREQuency → Query

Description Returns the gain margin frequency of PSRR results.

Syntax :POWer:PSRR:DATA:GMARgin:FREQuency?

Example :POWer:PSRR:DATA:GMARgin:FREQuency?

:POWer:PSRR:DATA:PMARgin → Query

Description Returns the phase margin of PSRR results.

Syntax :POWer:PSRR:DATA:PMARgin?

Example :POWer:PSRR:DATA:PMARgin?

:POWer:PSRR:DATA:PMARgin:FREQuency → Query

Description Returns the phase margin frequency of PSRR results.

Syntax :POWer:PSRR:DATA:PMARgin:FREQuency?

Example :POWer:PSRR:DATA:PMARgin:FREQuency?

:POWer:CLR:RUN

→ Set →

→ Query

Description Runs the control loop response function or returns the control loop response state.

Syntax :POWer:CLR:RUN  
:POWer:CLR:RUN?

Example :POWer:CLR:RUN

→ Set →

:POWer:CLR:STOP

→ Query

Description Stops the control loop response function or returns the control loop response state.

Syntax :POWer:CLR:STOP  
:POWer:CLR:STOP?

Example :POWer:CLR:STOP

→ Set →

:POWer:CLR:SOURce:INPut

→ Query

Description Sets or returns the input source for CLR.

Syntax :POWer:CLR:SOURce:INPut {CH1~CH4}  
:POWer:CLR:SOURce:INPut?

Parameter CH1~CH4 Channel 1 to Channel 4

Example :POWer:CLR:SOURce:INPut CH1  
Set the input source as channel 1.

:POWer:CLR:SOURce:OUTPut (Set) →  
→ (Query)

Description	Sets or returns the output source for CLR.
Syntax	:POWer:CLR:SOURce:OUTPut {CH1~CH4} :POWer:CLR:SOURce:OUTPut?
Parameter	CH1~CH4 Channel 1 to Channel 4
Example	:POWer:CLR:SOURce:OUTPut CH2 Set the output source as channel 2.

:POWer:CLR:FREQuency:STARt (Set) →  
→ (Query)

Description	Sets or returns the start frequency for CLR.
Syntax	:POWer:CLR:FREQuency:STARt {<Nrf>} :POWer:CLR:FREQuency:STARt?
Parameter	<Nrf> Sets the frequency to use. (Range:20Hz~25MHz)
Example	:POWer:CLR:FREQuency:STARt 100 Sets the start frequency as 100Hz.

:POWer:CLR:FREQuency:STOP (Set) →  
→ (Query)

Description	Sets or returns the stop frequency for CLR.
Syntax	:POWer:CLR:FREQuency:STOP {<Nrf>} :POWer:CLR:FREQuency:STOP?
Parameter	<Nrf> Sets the frequency to use. (Range:20Hz~25MHz)
Example	:POWer:CLR:FREQuency:STOP 500 Sets the stop frequency as 500Hz.




**:POWer:CLR:AWG:LOAD**

Description	Sets or returns the impedance for load.	
Syntax	:POWer:CLR:AWG:LOAD {FIFty HIGHZ} :POWer:CLR:AWG:LOAD?	
Parameter	FIFty	50 ohm
	HIGHZ	High impedance
Example	:POWer:CLR:AWG:LOAD HIGHZ Sets the load as high impedance.	


**:POWer:CLR:AWG:AMplitude**

Description	Sets or returns the amplitude for CLR. When amplitude profile is on, sets or returns the amplitude for the selected frequency.	
Syntax	:POWer:CLR:AWG:AMplitude {<NRf>,<range>} :POWer:CLR:AWG:AMplitude? [<range>]	
Parameter	<NRf>	Sets the amplitude to use.(Load=FIFTY, Range:0.01Vpp~2.5Vpp; Load=HIGHZ, Range:0.02Vpp~5Vpp)
	<range>	The selected frequency {F20hz   F100hz   F1Khz   F10Khz   F100Khz   F1Mhz   F10Mhz   F25Mhz} F20hz: Frequency range >20Hz (The default <range>). F100hz: Frequency range >100Hz. F1Khz: Frequency range >1kHz. F10Khz: Frequency range >10kHz. F100Khz: Frequency range >100kHz. F1Mhz: Frequency range >1MHz. F10Mhz: Frequency range >10MHz. F25Mhz: Frequency range 25MHz.

Example :POWer:CLR:AWG:AMPlitude 0.2  
 Sets the amplitude as 0.2Vpp.  
 :POWer:PSRR:CLR:AMPlitude 0.5,F100HZ  
 :POWer:PSRR:CLR:AMPlitude? F100HZ  
 0.5

Set →

:POWer:CLR:POINt

→ Query

Description Sets or returns the number of processing points in a decade.

Syntax :POWer:CLR:POINt {<NR1>}  
 :POWer:CLR:POINt?

Parameter <NR1> The number of points in a decade.(Range:10, 15, 30, 45, 90)

Example :POWer:CLR:POINt 15  
 Sets the number of processing points as 15 in a decade.

:POWer:CLR:SAVe

Set →

Description Saves the CLR result.

Syntax :POWer:CLR:SAVe

Example :POWer:CLR:SAVe  
 Saves the result to default file.

:POWer:CLR:RECAll

Set →

Description Recalls the CLR result from memory or USB.

Syntax :POWer:PSRR:RECAll {<file path>  
 ("Disk:/xxx.FRD", "USB:/xxx.FRD")}

Parameter xxx.FRD Filename

Example :POWer:CLR:RECALL "Disk:/LOP0002.FRD"  
 Recalls a CLR result named LOP0002.FRD from root directory (Disk:/) of the scope.

:POWer:CLR:DATA → Query

Description Shows the detailed information of CLR settings and results.

Syntax :POWer:CLR:DATA?

Example :POWer:CLR:DATA?  
 Shows the LCR result's detail.

:POWer:CLR:SAVETOCsv Set →

Description Saves the CLR result as a CSV file.

Syntax :POWer:CLR:SAVETOCsv

Example :POWer:CLR:SAVETOCsv  
 Saves results as CSV file.

:POWer:CLR:AWG:AMPlitude:PROFile Set →  
→ Query

Description Sets or returns the state of amplitude profile.

Syntax :POWer:CLR:AWG:AMPlitude:PROFile {ON|OFF}  
 :POWer:CLR:AWG:AMPlitude:PROFile?

Parameter	ON	Turn on the profile.
	OFF	Turn off the profile.

Example :POWer:CLR:AWG:AMPlitude:PROFile ON  
 :POWer:CLR:AWG:AMPlitude:PROFile?  
 ON

:POWer:CLR:AWG:AMPlitude:INTERPolation (Set) →  
→ (Query)

Description	Sets or returns the state of linear interpolation for the selected frequency.	
Syntax	:POWer:CLR:AWG:AMPlitude:INTERPolation {ON OFF,[<range>]} :POWer:CLR:AWG:AMPlitude:INTERPolation? [<range>]	
Parameter	ON	Turn on the interpolation.
	OFF	Turn off the interpolation.
	<range>	The selected frequency {F20hz   F100hz   F1Khz   F10Khz   F100Khz   F1Mhz   F10Mhz   F25Mhz} F20hz: Frequency range >20Hz (The default <range>). F100hz: Frequency range >100Hz. F1Khz: Frequency range >1kHz. F10Khz: Frequency range >10kHz. F100Khz: Frequency range >100kHz. F1Mhz: Frequency range >1MHz. F10Mhz: Frequency range >10MHz. F25Mhz: Frequency range 25MHz.
Example	:POWer:CLR:AWG:AMPlitude:INTERPolation ON,F100HZ :POWer:CLR:AWG:AMPlitude:INTERPolation? F100HZ ON	

:POWer:CLResponse:DATA:GMARgin → (Query)

Description	Returns the gain margin of CLR results.	
Syntax	:POWer:CLResponse:DATA:GMARgin?	
Example	:POWer: CLResponse:DATA:GMARgin?	

**:POWer:CLResponse:DATA:GMARgin:FREQuency** → **Query**

Description	Returns the gain margin frequency of CLR results.
Syntax	:POWer:CLResponse:DATA:GMARgin:FREQuency?
Example	:POWer:CLResponse:DATA:GMARgin:FREQuency?


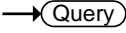
**:POWer:CLResponse:DATA:PMARgin** → **Query**

Description	Returns the phase margin of CLR results.
Syntax	:POWer:CLResponse:DATA:PMARgin?
Example	:POWer:CLResponse:DATA:PMARgin?

**:POWer:PSRR:DATA:PMARgin:FREQuency** → **Query**

Description	Returns the phase margin frequency of CLR results.
Syntax	:POWer:CLR:DATA:PMARgin:FREQuency?
Example	:POWer:CLR:DATA:PMARgin:FREQuency?

## USB Delay Command

:USBDelay		 
Description	Sets or returns the USB delay function for the PC connection which Windows 10 installed	
Syntax	:USBDelay {OFF ON} :USBDelay?	
Parameter/ Return parameter	<ON>	Turns on the USB delay function
	<OFF>	Turns off the USB delay function
Example	:USBDelay ON  Turns on the USB delay function when the scope connected with window 10 installed PC.	

## Digital Commands

:D<x>:DISPlay .....	415
:D<x>:POSition .....	415
:DISPlay:DIGital:HEIght .....	416
:DIGital:GROU<x>:THReshold .....	416
:DIGital:ANALog:A<x>:DISPlay .....	417
:DIGital:ANALog:A<x>:RATio .....	417
:DIGital:ANALog:A<x>:POSition .....	418
:D<x>:MEMory .....	418
:DIGital:MEMory .....	420

:D<x>:DISPlay (Set) →  
→ (Query)

**Description** Turns a digital channel <x> on/off or returns its status.

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

<b>Parameter / Return parameter</b>	OFF	Turns off a digital channel
	ON	Turns on a digital channel
	D<x>	Digital channel number D0 ~ D15

**Example** :D0:DISP OFF

:D<x>:POSition (Set) →  
→ (Query)

**Description** Sets or returns the position level for digital channel <x>.

**Syntax** :D<x>:POSition {<NRf> | ?}

<b>Parameter / Return parameter</b>	D<x>	Digital channel number D0 ~ D15
	<NRf>	Vertical scale position

Example :D0:POS ?  
 >1.87 DIV  
 :D0:POS 0

Set →

→ Query

**:DISPlay:DIGital:HEIght**

Description Sets or returns the number of available digital waveform position slots.

Syntax :DISPlay:DIGital:HEIght {SMALL | MEDium | LARge | ?}

Parameter / Return parameter	SMALL	Sets the height to small mode (digital channels: 16 max)
	MEDium	Sets the height to medium mode (digital channels: 16 max)
	LARge	Sets the height to large mode (digital channels: 8 max)

Example :DIS:DIG:HEI ?  
 >LARGE  
 :DIS:DIG:HEI SMA

Set →

→ Query

**:DIGital:GROU<x>:THReshold**

Description Sets or returns the digital threshold for a group.

Syntax :DIGital:GROU<x> {ECL | TTL | PECL | CMOS5 | CMOS3 | CMOS2 | <NR3> | ?}

Parameter / Return parameter	ECL	Sets the threshold to a preset ECL high level (-1.3V)
	TTL	Sets the threshold to a preset TTL high level (1.4V)
	PECL	Sets the threshold to a preset PECL high level (3.7V)
	CMOS5	Sets the threshold to a preset CMOS5 (5.0V) high level (2.5V)



CMOS3	Sets the threshold to a preset CMOS3 (3.3V) high level (1.65V)
CMOS2	Sets the threshold to a preset CMOS2 (2.5V) high level (1.25V)
<NR3>	Sets the threshold to a preset ECL high level (-1.3V)
GROUP<x>	Group number 1~4 (16 channels) or 1~2 (8 channels) GROUP1: digital channels D0~D3 GROUP2: digital channels D4~D7 GROUP3: digital channels D8~D11 GROUP4: digital channels D12~D15

Example :DIG:GROUP1:THR ?  
>-1.300e+00  
:DIG:GROUP1:THR TTL

Set →  
→ Query

**:DIGital:ANALog:A<x>:DISPlay**

Description Turns the analog waveform <x> on/off or returns its status.

Syntax :DIGital:ANALog:A<x>:DISPlay {OFF | ON | ?}

Parameter / Return parameter	OFF	Turns off the analog waveform
	ON	Turns on the analog waveform
	A<x>	Analog waveform number 1~2

Example :DIG:ANA:A1:DISP OFF

Set →  
→ Query

**:DIGital:ANALog:A<x>:RATio**

Description Sets or returns the analog waveform vertical scale ratio.

Syntax :DIGital:ANALog:A<x>:RATio {<NRf> | ?}

Parameter / Return parameter	<NRf> A<x>	Vertical scale ratio (0.1, 0.2, ...1). Analog waveform number 1~2
------------------------------	---------------	--

Example :DIG:ANA:A1:RAT 0.1

Set →  
 → Query

:DIGital:ANALog:A<x>:POSition

Description Sets or returns the analog waveform vertical scale position.

Syntax :DIGital:ANALog:A<x>:POSition {<NRf> | ?}

Parameter / Return parameter	<NRf> A<x>	Vertical scale position (0, 0.1, 0.2, ...8). Analog waveform number 1~2
------------------------------	---------------	--

Example :DIG:ANA:A1:POS 4.5

:D<x>:MEMory → Query

Description Returns the data in acquisition memory for the selected digital channel.

Syntax :D<x>:MEMory?

Related commands :ACQuire:RECOrdlength  
:HEADer

Parameter	D<x>	Digital channel number D0 ~ D15
Return parameter	<string>	Returns acquisition settings followed by raw waveform block data.  <string> Returns the acquisition settings for the selected digital channel.  Format: parameter(1),setting(1);parameter(2),setting(2)...parameter(n),setting(n); Waveform Data;

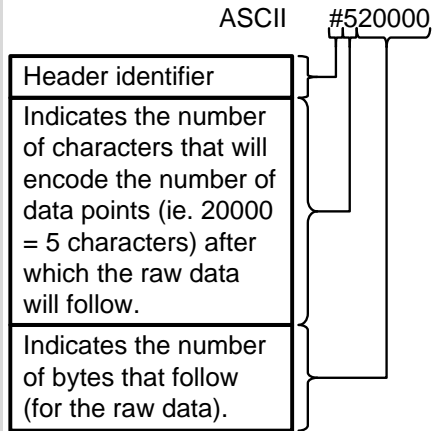
<waveform  
block data>

<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 logical level (0 or 1) of a data point of the digital channel, i.e. logical 0 is 0x0000 and logical 1 is 0x0001.

Waveform Raw Data Example:

Header raw data.....

Hex:

35 32 30 30 30 30 30 00 00 00 00 00 00  
00 01 00 01 .....

ASCII/Decimal:

#520000 0000000101 ....

The raw data contains 20000 bytes (=10000 points); point 1 is logical 0, point 2 is logical 0, point 3 is logical

0, point 4 is logical 1, point 5 is logical 1, etc...

Example :D1:MEM?  
 FORMAT,2.0A;Display,1;Memory  
 Length,10000;IntpDistance,0; Trigger Address,0;  
 Threshold Used,1.400E+00;Source,D1;Vertical  
 Units,V; Label1,;Firmware,V1.25b10; Horizontal  
 Units,S;Horizontal Scale,1.000E-04; Horizontal  
 Position,0.000E+00;Horizontal Mode,Main;SincET  
 Mode,Real Time; Sampling Period,1.000E-07;Time,  
 22-Sep-16 19:42:28;  
 Waveform Data;  
 #520000.....follows waveform block  
 data.....

**:DIGital:MEMory**



Description Returns the data in acquisition memory for all the digital channels.

Syntax :DIGital:MEMory?

Related commands :ACQuire:RECOrdlength  
 :HEADer

Return parameter Returns acquisition settings followed by raw waveform block data.

<string>

<string>

Returns the acquisition settings for all the digital channels.

Format:

parameter(1),setting(1);parameter(2),  
 setting(2)...parameter(n),setting(n);  
 Waveform Data;

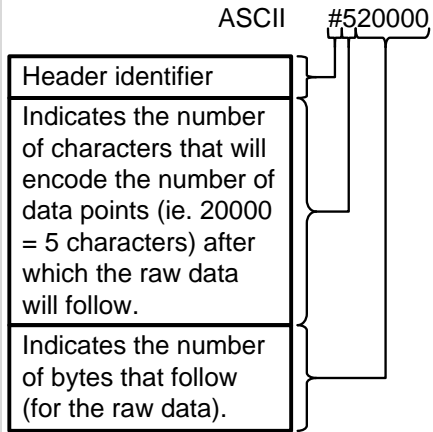
<waveform  
block data>

<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:

The sixteen bits composing each consecutive two bytes encode the logical level (0 or 1) of all the digital channels for one data point. For a given two bytes, the least significant bit is channel 0 and the most significant bit is channel 15.

Waveform Raw Data Example:

Header raw data.....

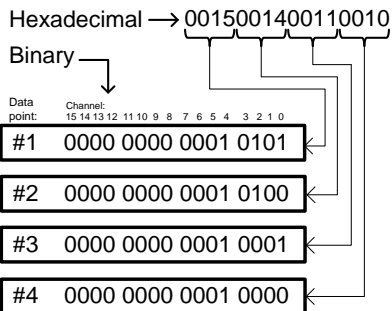
Hex:

35 32 30 30 30 30 30 00 15 00 14 00 11  
00 10 .....

ASCII/Decimal:

#520000 raw data ....

The raw data contains 20000 bytes (=10000 points) with the following logical level for each channels per data point:



Example

```
:DIG:MEM?
Format,2.0E;Display,1111111111111111;Memory
Length,10000;IntpDistance,0;Trigger
Address,0;Threshold12_15,1.40V;Threshold8_11,1.40
V;Threshold4_7,1.40V;Threshold0_3,1.40V;Vertical
Units,V;Label15,;Label14,;Label13,;Label12,;Label11,;L
abel10,;Label9,;Label8,;Label7,;Label6,;Label5,;Label4,
;Label3,;Label2,;Label1,;Label0,;Firmware,V1.25b10;H
orizontal Units,S;Horizontal Scale,1.000E-
04;Horizontal Position,0.000E+00;Horizontal
Mode,Main;SincET Mode,Real Time;Sampling
Period,1.000E-07;Time,22-Sep-16 10:44:28;
Waveform Data;
#520000.....follows waveform block
data.....
```

# APPENDIX

## Error messages

Description      The following error messages may be returned from the :SYSTem:ERRor? query. For details see page 186.

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 .....	56
ACQuire	
AVERAge .....	50
FILTer	
FREQuency .....	54, 55
FILTer.....	54
SOURce .....	53
MEMory.....	51
MODE.....	51
ACQuire	
STATe .....	56
ACQuire	
INTERpolation.....	57
ACQuire	
RECOrdlength .....	57
ACQuire	
HEADer .....	57
ACQuire	
HEADer .....	58
AUTORSET	
MODE.....	59
AUTOSet.....	59
BUS1 .....	198
CAN	
BITRate.....	209
PROBe .....	209
SAMPLepoint.....	209
SOURce .....	208
DISPlay	
FORMAt.....	208
I2C	
ADDResS	
RWINClude.....	199
SCLK	
SOURce .....	200
SDA	
SOURce .....	200, 201
INPut .....	199
LABel.....	280, 417
DISPlay ....	280, 281, 282, 283, 417, 420
LIN	
BITRate .....	210
IDFORmat .....	210
POLARity .....	211
SAMPLepoint .....	211
SOURce.....	211
STANDard .....	212
SPI	
BITORder .....	206
MISO	
SOURce .....	207
MOSI	
SOURce .....	207
SCLK	
POLARity .....	205
SOURce .....	206
SS	
POLARity .....	205
SOURce .....	206
WORDSize .....	205
STATE .....	198
TYPE .....	198
UART	
BITRate .....	202
DATABits .....	202
EOFPACKET .....	203
PACKet .....	203
PARItY .....	202
RX	
SOURce .....	204
TX	
SOURce .....	204
CHANnel	
BWLimit.....	60
COUPling.....	61
DESKew .....	61
DISPlay.....	61

EXPand .....	62	SAVE.....	327
IMPedance .....	62	SOURce .....	326
INVert .....	63	STATE .....	326
LAbel .....	277, 415	DATE .....	196
DISPlay.....	278, 415	DISPlay	
POStion.....	63	INTENSITY	
PROBe		BACKLight	
TYPE.....	64	AUTODim	
PROBe		Time.....	91
RATio .....	64	DISPlay	
SCALe.....	65	GRATicule.....	92
CLS .....	46	INTensity	
CURSor		BACKLight.....	91
DDT.....	80	BACKLight	
H1Position.....	81	AUTODim	
H2Position.....	81, 88	ENable .....	91
HDELta.....	82	GRATicule.....	90
HUNI .....	79	WAVEform.....	90
HUSE .....	79	OUTPut .....	93
MODE .....	78	PERSistence .....	92, 94
SOURce.....	78	WAVEform.....	93
V1Position.....	82	DVM	
VDELta .....	83	MODE.....	297
VUNI.....	80	SOURce.....	296
VUSE.....	80	STATE .....	296
XY		VALue .....	297
POLar		Error list .....	423
RADIUS		Ethernet	
DELta.....	85	interface.....	6
POStion .....	84	ETHERnet	
THETA		DHCP .....	192, 193, 194
DELta.....	85	FORCe .....	140
POStion .....	85	GONogo	
PRODuct		CLEar .....	298
DELta .....	86	EXECute216, 217, 218, 230, 231, 232,	
POStion.....	86	298, 305, 306, 307	
RATio		FUNction .....	218, 299, 308
DELta .....	87, 93	NGCount ....	218, 219, 220, 221, 222,
RATio		223, 224, 299, 308, 309, 310, 311,	
POStion.....	86	312, 313, 314, 315, 316, 317, 318,	
RECTangular		319, 320, 321, 322, 323, 344, 345,	
X		346, 347, 348, 414	
DELta.....	83	NGDefine.....	299
POStion .....	83	SCRipt .....	300
Y		SOURce .....	300
DELta.....	84		
POStion .....	84, 87, 88		
DATALOG			
DURation ...	328, 338, 339, 340, 341,		
342, 343			
INTerval .....	327		

VIOLation.....300  
 HARDcopy  
     ASSIGN.....98  
     MODE.....96  
     PRINTINKSaver.....97  
     SAVEFORMat.....97  
     SAVEINKSaver.....97  
     START.....96  
 HEADer ..... 57, 58  
 IDN? ..... 44  
 Interface ..... 5  
 LISTer  
     DATA.....208  
 LRN?..... 44  
 MARK ..... 213  
     CREATE.....213  
     DELEte.....214  
 MATH  
     ADVanced  
         OPERator .....73  
         POSition .....75  
         SCALe.....76  
     DISP.....66  
     DUAL  
         OPERator .....70  
         POSition .....70  
         SCALe.....70  
         SOURce .....69  
     FFT  
         HORizontal  
             POSition.....73  
             SCALe.....73  
         MAG.....71  
         POSition .....72  
         SCALe.....72  
         SOURce .....71  
         WINDow.....72  
     TYPE.....67  
 MATHVAR.....75  
 MEASure  
     AMPlitude.....113  
     AREa .....119  
     CARea .....119  
     CMEan .....114  
     CRMS.....118

FALL.....105  
 FFFDelay .....122  
 FFRDelay.....121  
 FOVShoot.....105  
 FPReshoot.....106  
 FREQuency .....107  
 FRFDelay.....121  
 FRRDelay .....120  
 GATing.....100  
 HIGH.....115  
 LFFDelay.....124  
 LFRDelay .....124  
 LOW .....115  
 LRFDelay .....123  
 LRRDelay.....122  
 MAX .....116  
 MEAN .....114  
 METHod ..... 101, 103, 104  
 MIN .....117  
 NEDGE.....113  
 NPULSE.....112  
 NWIDth.....107  
 PDUTy.....108  
 PEDGE.....112  
 PERiod.....108  
 PHAse.....125  
 PK2PK .....117  
 PPULSE.....111  
 PWIDth.....109  
 RISE.....109  
 RMS .....118  
 ROVShoot.....110  
 RPReshoot.....110  
 SOURce .....101  
 MEASUREMENT  
 MEAS  
     MAXimum .....130  
     MEAN.....131  
     MINimum .....131  
     SOURCE .....127  
     STATE.....129  
     STDdev .....132  
     TYPe.....128  
     VALue.....129  
 REFLevel  
     PERCent  
         HIGH.....102  
         LOW .....102  
         MID .....102  
         MID2 .....103  
 STATIstics.....133

MODE..... 133, 134, 135	BUS
WEIGHTing.....133	B1
Power Commands..... 355	CAN
RCL.....45, 234, 235, 236, 237	CONDition ..... 259
RECALL	DATA
SETUp.....187	QUALifier ..... 263
WAVEform .....187	SIZE.....264
REF	VALue ..... 264
DISPlay .....136	FRAMetype.....260
LABel ..... 278, 416	IDentifier
DISPlay..... 279, 416, 418	DIRection..... 262
OFFSet .....137	MODE..... 261
SCALe.....138	VALue ..... 261
TIMEbase	I2C
POSITION .....136	ADDRESS
SCALE.....137	DIRection..... 251
Remote control	MODE..... 249
interface configuration .....5	TYPE.....249
REMOTEDisk	VALue ..... 250
AUTOMount.....331	CONDition ..... 248
IPADDRESS .....329	DATA
MOUNT .....330	SIZE..... 251
PASSWord .....330	VALue ..... 252
PATHName .....329	LIN
USERName .....329	CONDition ..... 265
RST ..... 46	DATA
RUN ..... 139	QUALifier ..... 266
SAV ..... 45	SIZE.....267
SAVe	VALue .....267
IMAGe.....188	ERRTYpe.....268
FILEFormat.....188	IDentifier
INKSaver.....189	VALue ..... 268
SETUp.....189	SPI
WAVEform .....190	CONDition ..... 257
FILEFormat.....191	DATA
SEARCH	MISO..... 258
COPY ..... 240	MOSI.....259
FFTPeak	SIZE..... 257
METHod .....269	UART
MPEak.....270	CONDition ..... 253
SINFo.....270, 271, 272, 273, 274, 275	RX
STATE.....240	DATA .....253, 254
TOTAL.....240	TX
TRIGger	DATA ..... 256
	TYPe ..... 248
	EDGE
	SLOP ..... 241
	HLEVel ..... 243
	LEVel ..... 242
	LLEVel ..... 243
	PULSe
	TIME ..... 245
	WHEn..... 245
	PULSEWidth
	POLarity..... 243
	RISEFall
	SLOP..... 244

TIME.....	247	MAXimum .....	302
WHEn .....	247	MINimum .....	302
RUNT		SAVE	
POLarity .....	244	AUto .....	303
TIME.....	246	MAXimum .....	302
WHEn .....	246	MINimum .....	303
SOURce .....	241	TOLerance.....	303
TYPe .....	241		
SEGMENTS		TIMEbase	
CURRent.....	286	EXPand.....	141, 143
DISPALL.....	287	MODE .....	142
MEASure		POSition .....	141
MODE.....	288	SCALe.....	142
PLOT		WINDow	
DIVide.....	289	POSition.....	142
RESults.....	289	SCALe.....	143
SELect.....	289		
SOURce.....	288	TRIGger	
TABLE		ALTernate.....	159
LIST .....	291	BUS	
SAVE .....	292	B1	
SELect .....	291	CAN	
SOURce.....	290	CONDition .....	172
SAVE .....	292	DATA	
SELect		QUALifier.....	175
END .....	293, 294, 295	SIZE .....	176
START.....	293	VALue.....	176
SOURce .....	292	FRAMetype .....	173
STATE.....	285	IDentifier	
TIME.....	287	DIRection .....	174
TOTAlnum.....	286	MODE.....	173
		VALue.....	174
SET		I2C	
LABel .....	284, 418	ADDRess	
SINGLE .....	139	DIRection .....	165
Socket server		MODE.....	163
function check.....	10	TYPe .....	163
Socket server		VALue.....	164
interface .....	9	CONDition .....	162
STOP.....	139	DATA	
SYSTEM		SIZE .....	165
ERRor .....	186	VALue.....	166
LOCK.....	186	LIN	
TEMPlate		CONDition .....	177
MAXimum .....	301	DATA	
MINimum.....	301	QUALifier.....	178
MODE.....	301	SIZE .....	178
POSition		VALue.....	179
		ERRTYPe....	180, 181, 182, 183, 184
		IDentifier	
		VALue.....	180, 181
		SPI	
		CONDition .....	170
		DATA	
		MISO	

VALue .....	171	LEVel .....	149
MOSI		LLEVel .....	150
VALue .....	171	MODE .....	148
SIZE .....	170	NREJ .....	148
UART		PULSe	
CONDition .....	166	TIME .....	158
RX		WHEn .....	157
DATA		PULSEWidth	
SIZE .....	167	POLarity .....	152
VALue .....	168	RISEFall	
TX		SLOP .....	154
DATA		TIME .....	155
SIZE .....	168	WHEn .....	155
VALue .....	169	RUNT	
THReshold		POLarity .....	153
CH .....	162	TIME .....	154
TYPe .....	161	WHEn .....	153
COUPle .....	147	SOURce .....	147
DELay		STATe .....	160
EVENT .....	152	TIMEOut	
LEVel .....	152	TIMER .....	159
SLOP .....	151	WHEn .....	158
TIME .....	151	TYPe .....	146
TYPe .....	151	VIDeo	
EDGe		FIELD .....	156
SLOP .....	150	LINE .....	157
EXTERnal		POLarity .....	157
PRObe		TYPe .....	156
RATio .....	161		
TYPe .....	160		
FREQuency .....	146		
HLEVel .....	149		
HOLDoff .....	148		
		USB	
		remote control interface .....	5