

**DL9040/DL9140/DL9240 Series  
Digital Oscilloscope  
Communication Interface**

**U S E R ' S M A N U A L**

---

Thank you for purchasing the DL9040/DL9140/DL9240 Series Digital Oscilloscope (DL9040/DL9040L/DL9140/DL9140L/DL9240/DL9240L, hereafter referred to as the DL9000). This Communication Interface User's Manual describes the functions and commands of the following communication interfaces.

- USB Interface
- Ethernet Interface (Optional)
- GP-IB Interface

To ensure correct use, please read this manual thoroughly before beginning operation. After reading the manual, keep it in a convenient location for quick reference whenever a question arises during operation.

The following manuals are provided for the DL9040/DL9140/DL9240 series. Please read all of them.

Manual Title	Manual No.	Description
DL9040/DL9140/DL9240 Series Digital Oscilloscope User's Manual	IM 701310-01E	Explains all functions and procedures of the DL9040/DL9140/DL9240 Series excluding the communication functions.
DL9040/DL9140/DL9240 Series Digital Oscilloscope Communication Interface User's Manual (in CD)	IM 701310-17E	This manual. Explains the communication interface functions of the DL9040/DL9140/DL9240 Series.
DL9040/DL9140/DL9240 Series Digital Oscilloscope Serial Bus Signal Analysis Function User's Manual	IM 701310-51E	Explains the operating procedures of the optional I <sup>2</sup> C bus signal/CAN bus signal/LIN bus signal/SPI bus signal analysis function.
DL9000 Series Digital Oscilloscope/SB5000 Series Vehicle Serial Bus Analyzer Power Supply Analysis Function User's Manual	IM 701310-61E	Explains the operating procedures of the optional power supply analysis function.

## Notes

- This manual, IM 701310-17E 10th Edition, applies to DL9000 digital oscilloscope with firmware version 4.40 or later.  
If the most recent firmware version is not running on your DL9000, not all of the features described in this manual can be used.  
You can check the firmware version of your DL9000 on the overview screen.  
For instructions on how to open the overview screen, see section 18.4 in the *User's Manual IM 701310-01E*.  
To upgrade to the latest firmware version, go to the following Web page, and then browse to the download page.  
<http://tmi.yokogawa.com/service-support/downloads/>
- The contents of this manual are subject to change without prior notice as a result of continuing improvements to the instrument's performance and functions.
- Every effort has been made in the preparation of this manual to ensure the accuracy of its contents. However, should you have any questions or find any errors, please contact your nearest YOKOGAWA dealer.
- Copying or reproducing all or any part of the contents of this manual without the permission of Yokogawa Electric Corporation is strictly prohibited.

---

## USB Interface and Ethernet Interface

- The items below are needed on the PC to use the communication functions via the USB interface.
  - DL Series Library (TMCTL)
  - USB device driver for connecting the PC and the DL9000 series
- The items below are needed on the PC to use the communication functions via the Ethernet interface.
  - DL Series Library (TMCTL)

To download the libraries and drivers listed above, go to the following Web page, and then browse to the download page.

<http://tmi.yokogawa.com/service-support/downloads/>

## Sample Programs

To download sample programs, go to the following Web page, and then browse to the download page.

<http://tmi.yokogawa.com/service-support/downloads/>

## Trademarks

- Microsoft, MS-DOS, Visual C++, Windows, and Windows NT are registered trademarks of Microsoft Corporation in the United States and/or other countries.
- Adobe and Acrobat are trademarks of Adobe Systems Incorporated.
- For purposes of this manual, the TM and ® symbols do not accompany their respective trademark names or registered trademark names.
- Other company and product names are trademarks or registered trademarks of their respective holders.

## Revisions

1st Edition:	June 2005
2nd Edition:	September 2005
3rd Edition:	December 2005
4th Edition:	June 2006
5th Edition:	December 2006
6th Edition:	March 2007
7th Edition:	August 2007
8th Edition:	March 2008
9th Edition:	June 2008
10th Edition:	April 2009

---

# How to Use This Manual

## Structure of This Manual

This user's manual consists of the following sections.

**Chapter 1 Overview of the USB Interface**

Describes the functions and specifications of the USB interface.

**Chapter 2 Overview of the Ethernet Interface (Optional)**

Describes the functions and specifications of the Ethernet interface.

**Chapter 3 Overview of the GP-IB Interface**

Describes the functions and specifications of the GP-IB interface.

**Chapter 4 Before Programming**

Describes the syntax used to transmit commands.

**Chapter 5 Command**

Describes all the commands one by one.

**Chapter 6 Status Reports**

Describes the status byte, various registers, and queues.

**Appendix**

Describes reference material such as an ASCII character code table.

**Index**

Provides an index ordered alphabetically.

## Symbols and Notations Used in This Manual

### Safety Markings

The following markings are used in this manual.

---

**CAUTION** Calls attentions to actions or conditions that could cause light injury to the user or damage to the instrument or the user's data, and precautions that can be taken to prevent such occurrences.

**Note** Calls attention to information that is important for proper operation of the instrument.

### Notation Used in the Procedural Explanations

On pages that describe the operating procedures in chapters 1 through 3, the following notations are used to distinguish the procedures from their explanations.

**Procedure** Carry out the procedure according to the step numbers. All procedures are written with inexperienced users in mind; experienced users may not need to carry out all the steps.

**Explanation** This section describes the setup items and the limitations regarding the procedures.

### Notation of User Controls

---

#### Operation/Soft Key Names and Menu Items Set in Boldface

Boldface type indicates the names of user-controlled operation keys on the instrument panel, and soft key items and menu items displayed on screen.

---

#### SHIFT+Panel Key

The SHIFT+Panel key means you will press the SHIFT key to turn ON the indicator of SHIFT key and then press the panel key. The menu marked in purple above the pressed key appears on the screen.

---

### Unit

---

k: Denotes "1000." Example: 100 kS/s (sample rate)

K: Denotes "1024." Example: 720 KB (file data size)

---

## Symbols Used in the Syntax

The following table indicates symbols that are used in the syntax mainly in chapters 4 and 5. These symbols are referred to as BNF (Backus-Naur Form) symbols. For details on the data, see pages 4-5 and 4-6.

Symbol	Meaning	Example	Example of Input
	Defined value	CHANnel<x> <x> = 1 to 4	CHANNEL2
{ }	Select from values given in { }	COUPling {AC DC DC50 GND}	COUPLING AC
	Exclusive OR		
[ ]	Can be omitted	TRIGger [:SIMPlE]:SLOPe	TRIGger:SLOPe

---

# Contents

How to Use This Manual.....	iii
<b>Chapter 1 Overview of the USB Interface</b>	
1.1 Part Names and Functions.....	1-1
1.2 USB Interface Functions and Specifications.....	1-2
1.3 Connection via the USB Interface.....	1-3
1.4 Setting the DL9000 (USB).....	1-4
<b>Chapter 2 Ethernet Interface</b>	
2.1 Part Names and Functions.....	2-1
2.2 Ethernet/VXI-11 Interface Features and Specifications.....	2-2
2.3 Connecting the Ethernet/VXI-11 Interface.....	2-4
2.4 Setting the DL9000 (Network).....	2-5
<b>Chapter 3 Overview of the GP-IB Interface</b>	
3.1 Part Names and Functions.....	3-1
3.2 Connecting the GP-IB Card.....	3-2
3.3 GP-IB Interface Functions.....	3-3
3.4 GP-IB Interface Specifications.....	3-4
3.5 Setting the DL9000 (GP-IB).....	3-5
3.6 Responses to Interface Messages.....	3-6
<b>Chapter 4 Before Programming</b>	
4.1 Messages.....	4-1
4.2 Commands.....	4-3
4.3 Response.....	4-5
4.4 Data.....	4-6
4.5 Synchronization with the Controller.....	4-8
<b>Chapter 5 Commands</b>	
5.1 List of Commands.....	5-1
5.2 ACQUIRE Group.....	5-34
5.3 ANALYSIS Group.....	5-35
5.4 ASETUP Group.....	5-65
5.5 CALIBRATE Group.....	5-66
5.6 CHANNEL Group.....	5-67
5.7 CLEAR Group.....	5-69
5.8 COMMUNICATE Group.....	5-70
5.9 CURSOR Group.....	5-72
5.10 DISPLAY Group.....	5-82
5.11 FILE Group.....	5-85
5.12 GONOGO Group.....	5-91
5.13 HCOPY Group.....	5-100
5.14 HISTORY Group.....	5-102
5.15 IMAGE Group.....	5-112
5.16 INITIALIZE Group.....	5-113
5.17 MATH Group.....	5-114
5.18 MEASURE Group.....	5-122
5.19 REFERENCE Group.....	5-131

5.20	SEARch Group .....	5-132
5.21	SNAP Group .....	5-137
5.22	SSTart Group .....	5-138
5.23	STARt Group .....	5-139
5.24	STATus Group .....	5-140
5.25	STOP Group .....	5-141
5.26	SYSTem Group .....	5-142
5.27	TELeomtest Group .....	5-143
5.28	TIMEbase Group .....	5-148
5.29	TRIGger Group .....	5-149
5.30	WAVEform Group .....	5-200
5.31	ZOOM Group .....	5-202
5.32	Common Command Group .....	5-204

**Chapter 6 Status Reports**

6.1	Overview of the Status Report .....	6-1
6.2	Status Byte .....	6-3
6.3	Standard Event Register .....	6-4
6.4	Extended Event Register .....	6-5
6.5	Output Queue and Error Queue .....	6-6

**Appendix**

Appendix 1	ASCII Character Codes .....	App-1
Appendix 2	Error Messages .....	App-2
Appendix 3	Waveform Parameter Name Table .....	App-4
Appendix 4	Eye Pattern Parameter Name Table .....	App-5

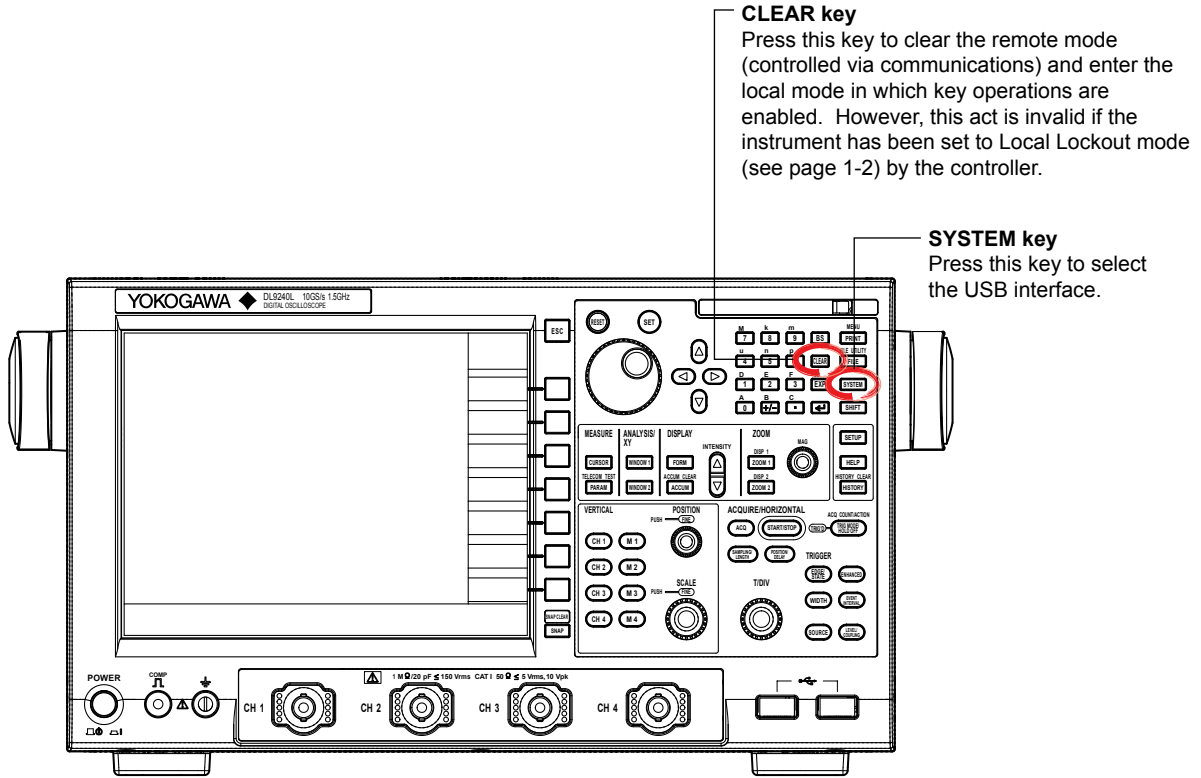
**Index**



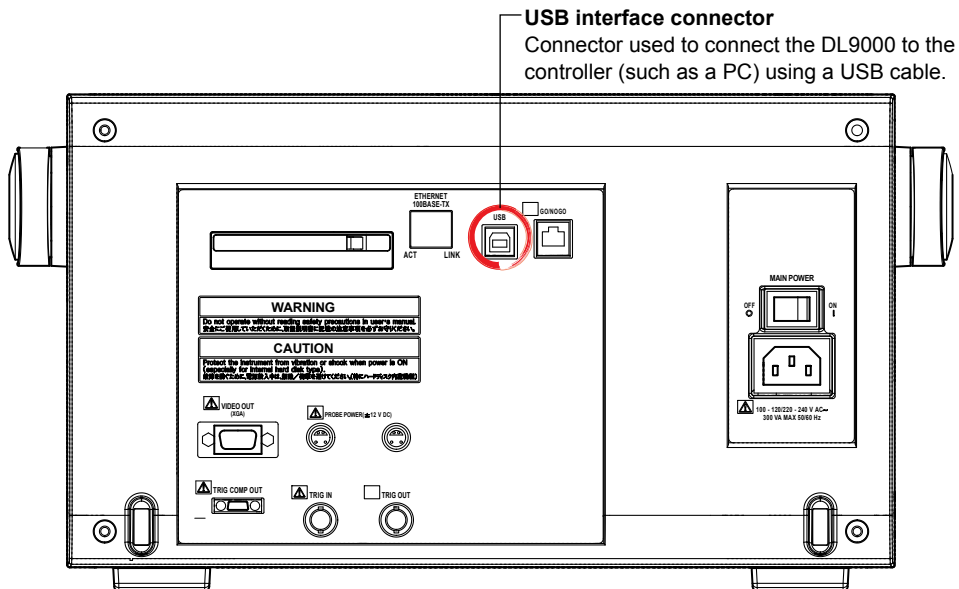


# 1.1 Part Names and Functions

## Front Panel



## Rear Panel



## 1.2 USB Interface Functions and Specifications

### USB Interface Functions

#### Reception Function

You can specify the same settings as those specified by front panel key operations. Receives output requests for measured and computed data, setup parameters of the panel, and error codes.

#### Transmission Function

Outputs measured and computed data.  
Outputs panel setup parameters and the status byte.  
Outputs error codes that have occurred.

### USB Interface Specifications

Electrical and mechanical specifications: Conforms to USB Rev. 2.0

Connector: Type B connector (receptacle)

Number of ports: 1

Power supply: Self-powered

PC system supported: PC running Windows 2000 or Windows XP with a standard USB port (a separate device driver is needed to connect to a PC).

### Data Rate

The response time when outputting waveform data is shown below as a reference.

Model: 701313 main unit

Controller: PC (Pentium4 3.4 GHz, USB2.0) and OS (Windows XP Professional SP1)

Language used: Visual C++

Number of Data Points	Word Data	ASCII Data
2500	Approx. 51 ms	Approx. 0.469 s
125000	Approx. 193 ms	Approx. 22.766 s
1250000	Approx. 1606 ms	Approx. 224.890 s
2500000	Approx. 3188 ms	Approx. 451.297 s
6250000	Approx. 7841 ms	Approx. 1127.625 s

### Switching between Remote and Local Modes

#### When Switching from Local to Remote Mode

If the DL9000 receives a “:COMMunicate:REMOte ON” command from the PC when it is in the local mode, it switches to the remote mode.

- REMOTE is displayed in the center of the upper section of the screen.
- All keys except the **CLEAR** key are disabled.
- Settings entered in local mode are retained even when the DL9000 switches to remote mode.

#### When Switching from Remote to Local Mode

Pressing **CLEAR** in remote mode puts the instrument in local mode. However, this is void when the DL9000 has received a “:COMMunicate:LOCKout ON” command from the PC (local lockout condition). When the DL9000 receives a “:COMMunicate:REMOte OFF” command from the PC, the DL9000 switches to the local mode regardless of the local lock condition.

- REMOTE indication in the center of the upper section of the screen disappears.
- Key operations are enabled.
- Settings entered in remote mode are retained even when the DL9000 switches to local mode.

#### **Note**

The USB interface cannot be used simultaneously with another interface (GP-IB or Ethernet interface).

---

## 1.3 Connection via the USB Interface

### Precautions to Be Taken When Making Connections

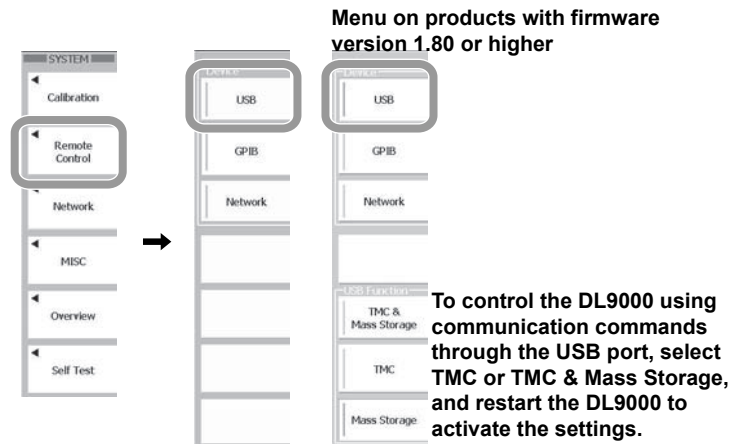
- Connect the USB cable by inserting the connector firmly into the USB connector.
- When connecting multiple devices using USB hubs, connect the DL9000 to the USB hub that is closest to the controller.
- Do not insert the USB cable into the GO/NO-GO output terminal. If you do, the instrument may malfunction.

## 1.4 Setting the DL9000 (USB)

### Procedure

#### Selecting the Communication Interface

1. Press **SYSTEM**.
2. Press the **Remote Control** soft key.
3. Press the **USB** soft key.

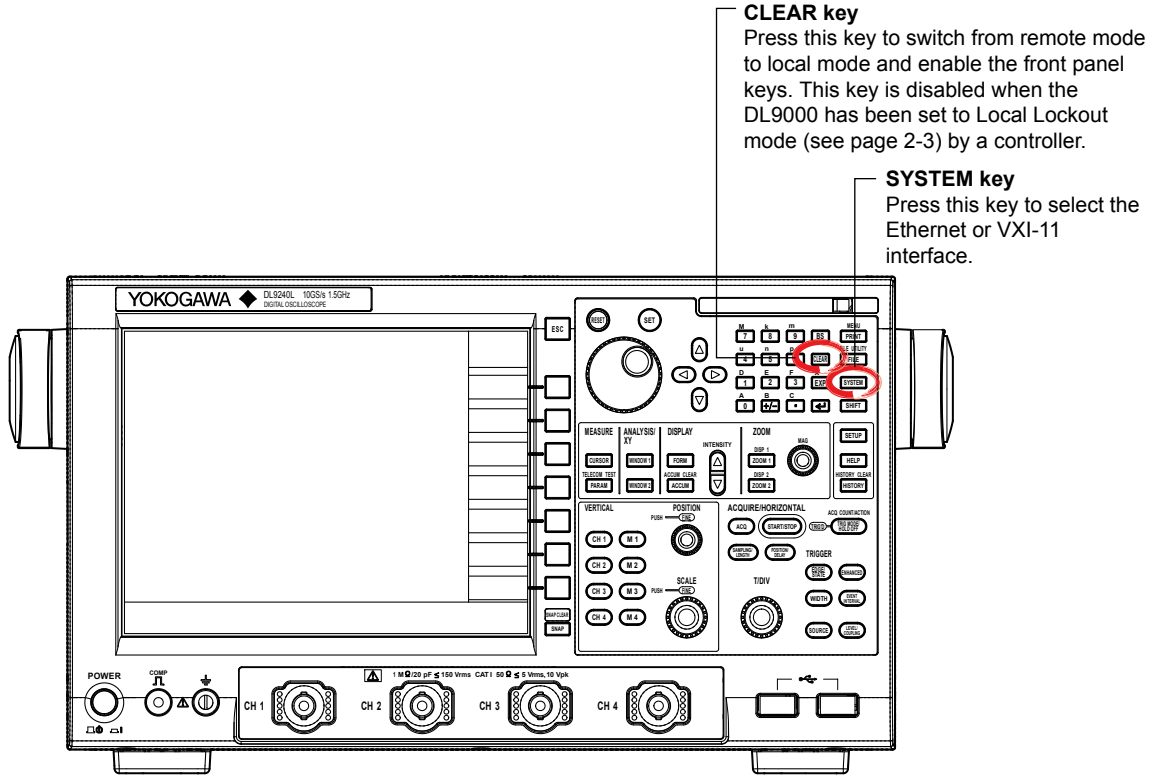


#### Note

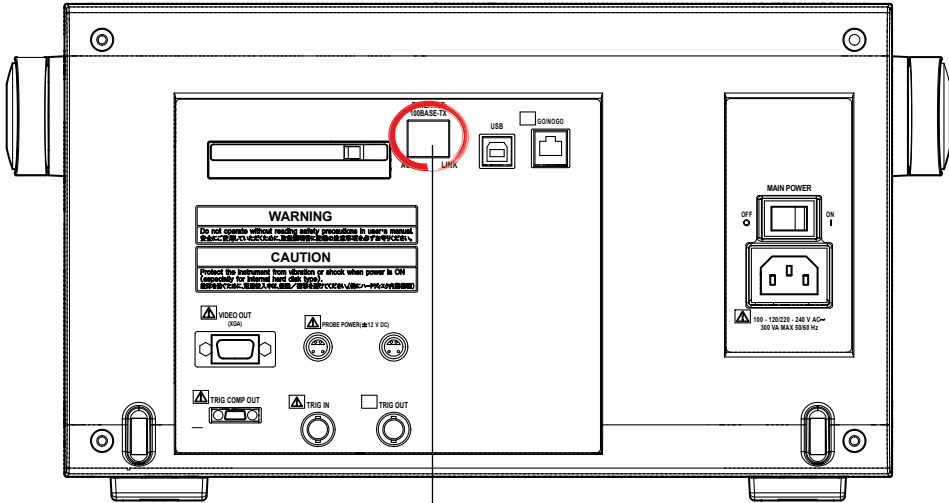
- Only the communication interface selected by Remote Control is enabled. The DL9000 does not accept commands that are transmitted to other unselected communication interfaces.
- To control the DL9000 remotely using communication commands through the USB port, select "TMC" in the menu above, and carry out the procedure below.
  - You must restart the DL9000 to activate the TMC, Mass Storage, or TMC & Mass Storage setting. Wait at least 10 seconds after you turn the power switch OFF, and then turn the switch back ON.
  - Install YOKOGAWA's TMC (Text and Measurement Class) driver into your PC. To obtain YOKOGAWA's USB TMC driver, contact your nearest YOKOGAWA dealer or access the following USB driver page at our Web site and download it.  
<http://www.yokogawa.com/tm/tm-softdownload.htm>
  - Only use the USB TMC driver (or software) provided by YOKOGAWA.
- The DL9000 can be controlled remotely even when TMC & Mass Storage is enabled and the DL9000 is connected to the PC. However, file operation is not possible using communication commands. In addition, file operation is also not possible using the keys on the DL9000. Disconnect the PC or activate the TMC setting and connect the DL9000 to the PC. File operation is also not possible using keys on the DL9000 when the Mass Storage setting is enabled.

# 2.1 Part Names and Functions

## Front Panel

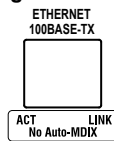


## Rear Panel



**Ethernet port**  
This port is for connecting the DL9000 to a controller (such as a PC) using an Ethernet cable.

**Marking on models with the LXI option (/C9 and /C12)**



## 2.2 Ethernet/VXI-11 Interface Features and Specifications

### Ethernet/VXI-11 Interface Features

#### Reception Feature

The DL9000 reception feature allows you to specify the same settings through an Ethernet connection that you can specify using the front panel keys.

The DL9000 can receive output requests for measured and computed data, panel setting data, and error codes.

#### Transmission Feature

The DL9000 can transmit measured and computed data.

The DL9000 can transmit panel setting data and the status byte.

The DL9000 can transmit error codes when errors occur.

### Ethernet/VXI-11 Interface Specifications

Electrical and mechanical specifications	IEEE802.3 compliant	
Simultaneous connections	1	
Port number	Ethernet:	10001/tcp
	VXI-11:	10240/tcp, 10250/tcp, 111/tcp, and 111/udp

### Data Transfer Rates

The response time when outputting waveform data is shown below as a reference.

Model: 701313 main unit

Controller: PC (Pentium4 3.4 GHz) and OS (Windows XP Professional SP1)

Network adapter: Corega FEther PCI-TXL

Language used: Visual C++

#### Ethernet Interface (/C8, /C9, /C10 and /C12 options)

Number of Data Points	Word Data	ASCII Data
2500	Approx. 16 ms	Approx. 0.391 s
125000	Approx. 259 ms	Approx. 19.063 s
1250000	Approx. 2313 ms	Approx. 189.812 s
2500000	Approx. 4595 ms	Approx. 379.750 s
6250000	Approx. 10400 ms	Approx. 950.532 s

#### VXI-11 Ethernet Interface (/C8, /C9, /C10 and /C12 options)

Number of Data Points	Word Data	ASCII Data
2500	Approx. 31 ms	Approx. 0.406 s
125000	Approx. 235 ms	Approx. 19.172 s
1250000	Approx. 2286 ms	Approx. 191.876 s
2500000	Approx. 4541 ms	Approx. 383.012 s
6250000	Approx. 11150 ms	Approx. 958.128 s

## Switching between Remote and Local Modes

### When Switching from Local to Remote Mode

The DL9000 switches to remote mode when it is in local mode and it receives a `:COMMunicate:REMOte ON` command from the PC.

- “REMOTE” appears at the top center of the screen.
- All keys except the **CLEAR** key are disabled.
- Settings entered in local mode are retained even when the DL9000 switches to remote mode.

### When Switching from Remote to Local Mode

When the DL9000 is in remote mode and you press **CLEAR**, the DL9000 switches to Local mode, unless it has received a `:COMMunicate:LOCKout ON` command from the PC, in which case pressing **CLEAR** will have no effect. The DL9000 switches to local mode when it receives a `:COMMunicate:REMOte OFF` command from the PC, regardless of the local lockout state.

- The “REMOTE” indicator at the top center of the screen disappears.
- Key operations are enabled.
- Settings entered in remote mode are retained even when the DL9000 switches to local mode.

### **Note**

The Ethernet/VXI-11 interface cannot be used simultaneously with another interface (USB or GP-IB interface).

## User Authentication Feature

You must enter a user name and password to access the DL9000 through the Ethernet interface. If the SB Series Library (TMCTL) version is 1.40 or later, the password is encrypted using the MD5 algorithm (RSA Data Security, Inc. MD5 Message-Digest Algorithm) and then sent to the DL9000.

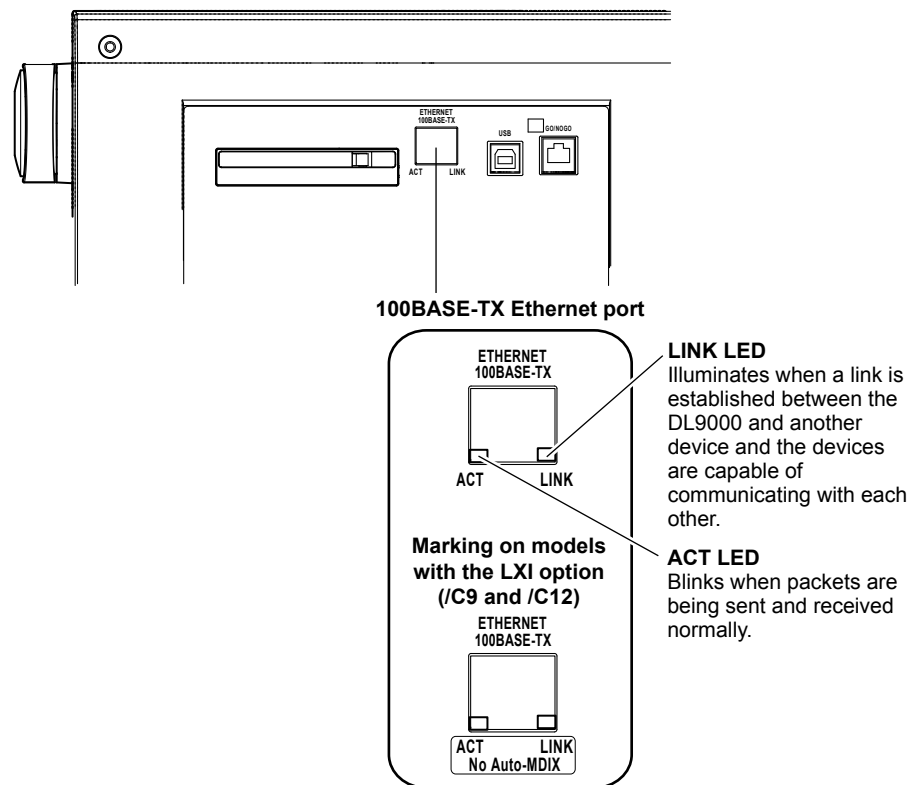
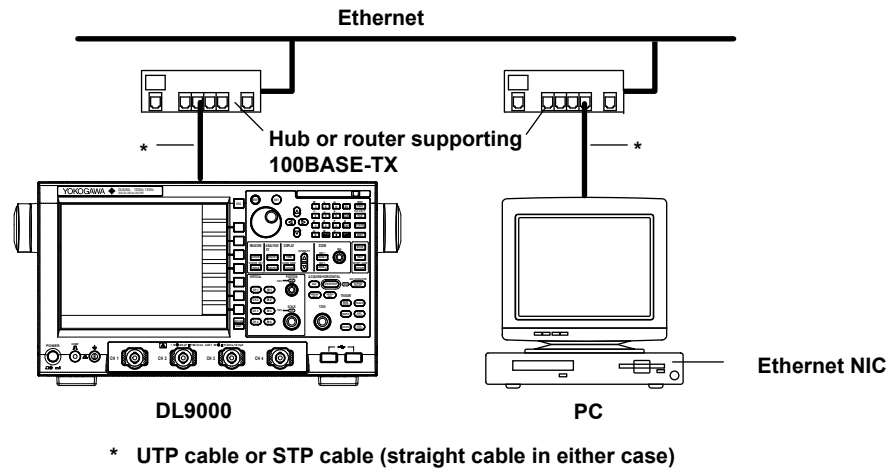
Set the user name and password in the Remote Control setup screen in the DL9000 SYSTEM menu. For instructions on how to set the user name and password, see section 2.4, “Setting the DL9000 (Network).”

If you select VXI-11 in the Device menu of the Remote Control setup screen (explained later), the user authentication setup menu will not appear in the Remote Control setup screen.

## 2.3 Connecting the Ethernet/VXI-11 Interface

### Connection Procedure

Connect a UTP (Unshielded Twisted-Pair) or STP (Shielded Twisted-Pair) cable that is connected to a hub or other network device to the 100BASE-TX port on the DL9000 rear panel.



### Notes about Connections

- Be sure to connect the DL9000 to a PC through a hub using straight cables. The DL9000 may not operate properly if you connect it to a PC directly using a cross cable.
- If you use UTP cables (straight cables), make sure that they are category 5.

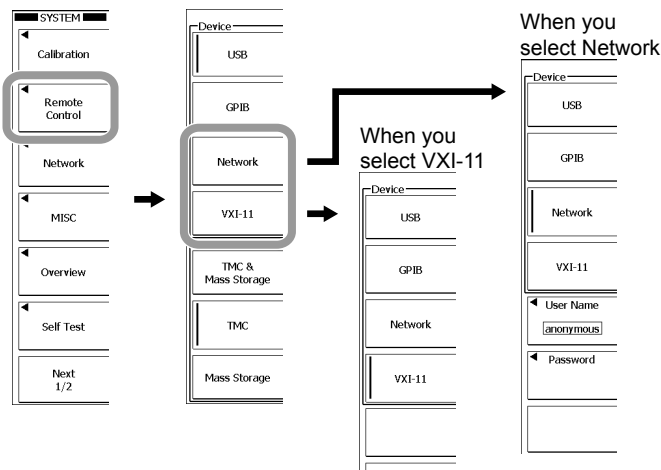


## 2.4 Setting the DL9000 (Network)

### Procedure

#### Selecting the Communication Interface

1. Press **SYSTEM**.
2. Press the **Remote Control** soft key. The Device menu appears.
3. Press the **Network** or **VXI-11** soft key.
  - If you want to use the Ethernet interface, press Network.
  - If you want to use the VXI-11 interface, press VXI-11.



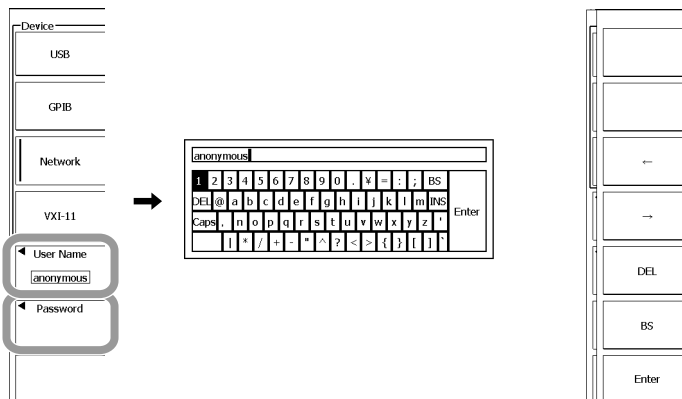
#### Note

Only the interface that you select after pressing Remote Control is valid. The DL9000 does not accept commands that are transmitted to other unselected communication interfaces.

#### User Name and Password

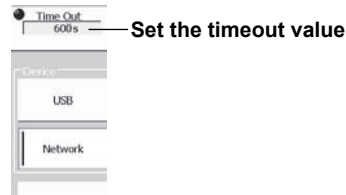
(This setup menu does not appear when you select VXI-11 in step 3.)

4. Press the **UserName** soft key. A keyboard appears on the screen.
5. To enter the user name, follow the procedure in section 4.2 of the *DL9000 User's Manual (IM701310-01E)*.
6. Enter the password in the same manner that you entered the user name.



### Setting the Timeout Value

7. Use the **rotary knob** to set the timeout value.



---

### Note

- For details on how to operate the soft keyboard, see section 4.2, “Entering Values and Strings” in the *DL9000 User’s Manual (IM701310-01E)*.
  - User names and passwords are case-sensitive.
- 

### Configuring TCP/IP Settings

To use the Ethernet interface, you must specify the following TCP/IP settings.

- IP address
- Subnet mask
- Default gateway

For details on how to specify these settings, see section 15.2, “Setting Up the TCP/IP” in the *DL9000 User’s Manual (IM701310-01E)*.

### Explanation

To use a controller to configure the same settings that you can configure using the DL9000 keys or to send setting parameters and waveform data to a controller, you must configure the settings described below.

#### User Name and Password

The Ethernet interface has a user authentication feature. Set the DL9000 user name and password in advance.

If you select VXI-11 in the Device menu of the Remote Control setup screen, the user authentication setup menu will not appear in the Remote Control setup screen.

- **User Name**  
Set the user name using up to 30 characters. The default setting is “anonymous.”
- **Password**  
Set the password using up to 30 characters.

#### Timeout Value

The DL9000 disconnects itself from the controller if it is not accessed for the period of time that you specify using the timeout value.

#### TCP/IP

For details, see section 15.2, “Setting Up the TCP/IP” in the *DL9000 User’s Manual (IM701310-01E)*.

---

### Note

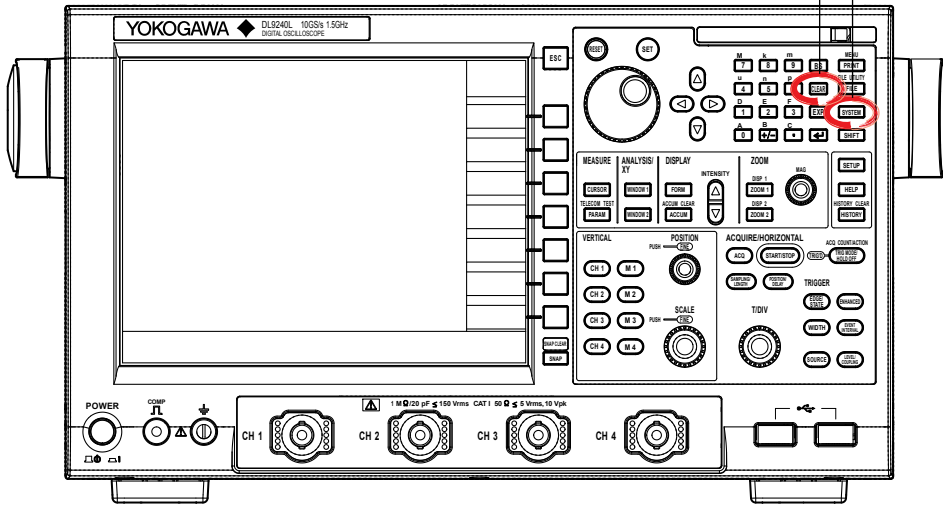
- The DL9000 terminates the connection if there is an error in the user authentication process.
  - You do not need to enter a password when the user name is set to “anonymous.”
-

# 3.1 Part Names and Functions

## Front Panel

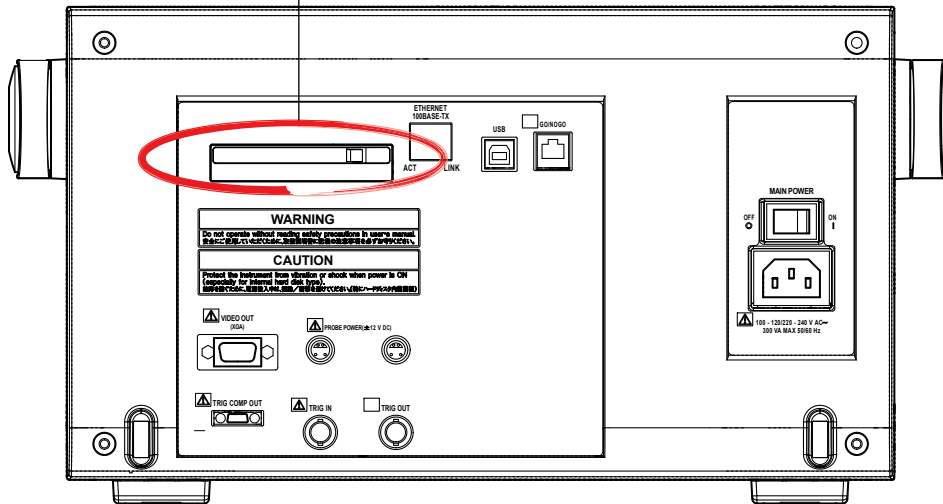
**CLEAR key**  
Press this key to clear the remote mode (controlled via communications) and enter the local mode in which key operations are enabled. However, this act is invalid if the instrument has been set to Local Lockout mode (see page 3-3) by the controller.

**SYSTEM key**  
Press this key to select the GP-IB interface.



## Rear Panel

**PC card slot**  
Connect a GP-IB card to the PC card slot and connect to a controller (such as a PC).



## 3.2 Connecting the GP-IB Card

### GP-IB Card

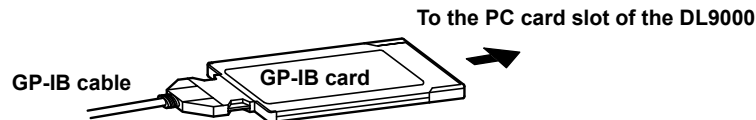
Use the NI PCMCIA-GPIB card by National Instruments.

### GP-IB Cable

Use the GP-IB cable that comes with the GP-IB card.

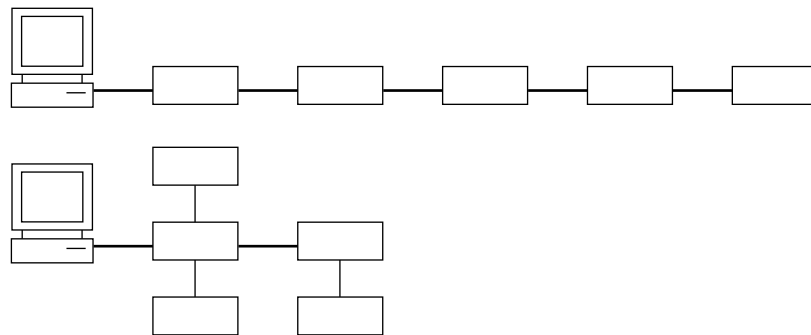
### Connection Procedure

Insert the GP-IB card into the DL9000 PC card slot.



### Precautions to Be Taken When Making Connections

- For the handling of the GP-IB card, see the manual that comes with the GP-IB card.
- Attach the GP-IB correctly with the front face up.
- Connect the GP-IB card to the DL9000 PC card slot first. Then, turn the DL9000 ON.
- Multiple cables can be used to connect multiple devices. However, no more than 15 devices including the controller can be connected on a single bus.
- When connecting multiple devices, each device must have its own unique address.
- Use a cable of length 2 m or less for connecting the devices.
- Make sure the total cable length does not exceed 20 m.
- When communicating, have at least two-thirds of the devices turned ON.
- When connecting multiple devices, connect them in a star or linear configuration (see the figure below). Do not wire them in a loop or parallel configuration.



### CAUTION

When connecting or disconnecting communication cables, make sure to turn OFF the PC and the DL9000. Otherwise, erroneous operation or damage to the internal circuitry may result.

## 3.3 GP-IB Interface Functions

### GP-IB Interface Functions

#### Listener Capability

- All of the information that you can set with the panel keys can be set through the GP-IB interface except for turning ON/OFF the power and setting the communication parameters.
- Receives commands from a controller requesting the output of setup information, waveform data, and other information.
- Also receives status report commands.

#### Talker Capability

Outputs setup information, waveform data, and other information.

#### **Note**

---

Listen-only, talk-only, and controller capabilities are not available on the DL9000.

---

### Switching between Remote and Local Modes

#### When Switching from Local to Remote Mode

Receiving a REN (Remote Enable) message from the PC when the DL9000 is in the local mode causes the DL9000 to switch to the remote mode.

- REMOTE is displayed in the center of the upper section of the screen.
- All keys except the **CLEAR** key are disabled.
- Settings entered in local mode are retained even when the DL9000 switches to remote mode.

#### When Switching from Remote to Local Mode

Pressing **CLEAR** in remote mode puts the DL9000 in local mode. However, this act is invalid if the DL9000 has been set to Local Lockout mode (see page 3-6) by the controller.

- REMOTE indication in the center of the upper section of the screen disappears.
- Key operations are enabled.
- Settings entered in remote mode are retained even when the DL9000 switches to local mode.

#### **Note**

---

The GP-IB interface cannot be used simultaneously with another interface (USB or Ethernet interface).

---

## 3.4 GP-IB Interface Specifications

### GP-IB Interface Specifications

Electrical and mechanical specifications:	Conforms to IEEE St'd 488-1978
Functional specifications:	See table below.
Protocol:	Conforms to IEEE St'd 488.2-1992
Code used:	ISO (ASCII) code
Mode:	Addressable mode
Address setting:	The address can be set in the range from 0 to 30 on the GP-IB setting screen that is played using the MISC menu.
Clear remote mode:	Remote mode can be cleared by pressing <b>CLEAR</b> except when the DL9000 has been set to Local Lockout mode by the controller.

### Functional specifications

Function	Subset Name	Description
Source handshaking	SH1	Full source handshaking capability
Acceptor handshaking	AH1	Full acceptor handshaking capability
Talker	T6	Basic talker capability, serial polling, untalk on MLA (My Listen Address), and no talk-only capability
Listener	L4	Basic listener capability, unlisten on MTA (My Talk Address), and no listen-only capability.
Service request	SR1	Full service request capability
Remote local	RL1	Full remote/local capability
Parallel polling	PP0	No parallel polling capability
Device clear	DC1	Full device clear capability
Device trigger	DT0	No device trigger capability
Controller	C0	No controller capability
Electrical characteristics	E1	Open collector

### Data Rate

The response time when outputting waveform data is shown below as a reference.

Model: 701313  
 Controller: PC (Pentium4 3.4 GHz, USB 2.0), OS (Windows XP Professional SP1)  
 Language used: Visual C++

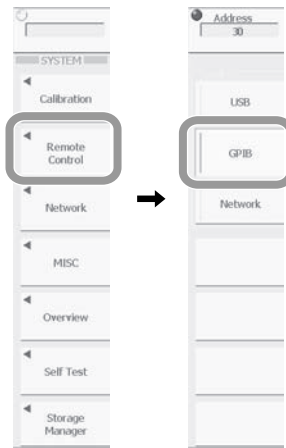
Number of Data Points	Word Data	ASCII Data
2s500	Approx. 16 ms	Approx. 0.390 s
125000	Approx. 344 ms	Approx. 19.453 s
1250000	Approx. 3172 ms	Approx. 194.516 s
2500000	Approx. 6282 ms	Approx. 389.047 s
6250000	Approx. 15641 ms	Approx. 971.985 s

## 3.5 Setting the DL9000 (GP-IB)

### Procedure

#### Selecting the Communication Interface

1. Press **SYSTEM**.
2. Press the **Remote Control** soft key to display the Device menu.
3. Press the **GP-IB** soft key.



#### Note

Only the communication interface selected by Remote Control is enabled. The DL9000 does not accept commands that are transmitted to other unselected communication interfaces.

#### Setting the Address

4. Turn the **rotary knob** to set the address.



### Explanation

Enter the following settings when using a controller to set information that can be specified through key operation on the DL9000 or when outputting setting parameters or output waveform data to the controller.

#### Setting the Address

Set the address of the DL9000 within the following range for the addressable mode.  
0 to 30

Each device that can be connected via GP-IB has a unique address within the GP-IB system. This address is used to distinguish the device from others. Therefore, when you connect the DL9000 to a PC, for example, make sure to assign a unique address to the DL9000.

#### Note

Do not change the address while the controller is communicating with the DL9000 or other devices over the GP-IB.

---

## 3.6 Responses to Interface Messages

### Responses to Interface Messages

#### Responses to a Uni-Line Message

- **IFC (Interface Clear)**  
Clears the talker and listener functions. Stops output if data are being output.
- **REN (Remote Enable)**  
Switches between the remote and local modes.  
  
IDY (Identify) is not supported.

#### Responses to a Multi-Line Message (Address Command)

- **GTL (Go To Local)**  
Switches to the local mode.
- **SDC (Selected Device Clear)**
  - Clears the program message (command) being received and the output queue (see page 6-5).
  - \*OPC and \*OPC? commands in execution are void.
  - The \*WAI and COMMunicate:WAIT commands are immediately terminated.  
PPC (Parallel Poll Configure), GET(Group Execute Trigger), and TCT (Take Control) are not supported.

#### Responses to a Multi-Line Message (Universal Command)

- **LLO (Local Lockout)**  
Disables **CLEAR** on the front panel to prohibit switching to the local mode.
- **DCL (Device Clear)**  
Same operation as the SDC message.
- **SPE (Serial Poll Enable)**  
Sets the talker function on all devices on the bus to serial polling mode. The controller polls the devices in order.
- **SPD (Serial Poll Disable)**  
Clears the serial polling mode of the talker function on all devices on the bus.  
  
PPU (Parallel Poll Unconfigure) is not supported.

### What Are Interface Messages

Interface messages are also referred to as interface commands or bus commands. They are commands that are issued by the controller. They are classified as follows:

#### Uni-Line Messages

A single control line is used to transmit uni-line messages. The following three types are available.

- IFC (Interface Clear)
- REN (Remote Enable)
- IDY (Identify)



### Multi-Line Messages

Eight data lines are used to transmit multi-line messages. The messages are classified as follows:

- **Address Commands**

These commands are valid when the instrument is designated as a listener or as a talker. The following five types are available.

Commands that are valid on an instrument that is designated as a listener

- GTL (Go To Local)
- SDC (Selected Device Clear)
- PPC (Parallel Poll Configure)
- GET (Group Execute Trigger)

Commands that are valid on an instrument that is designated as a talker

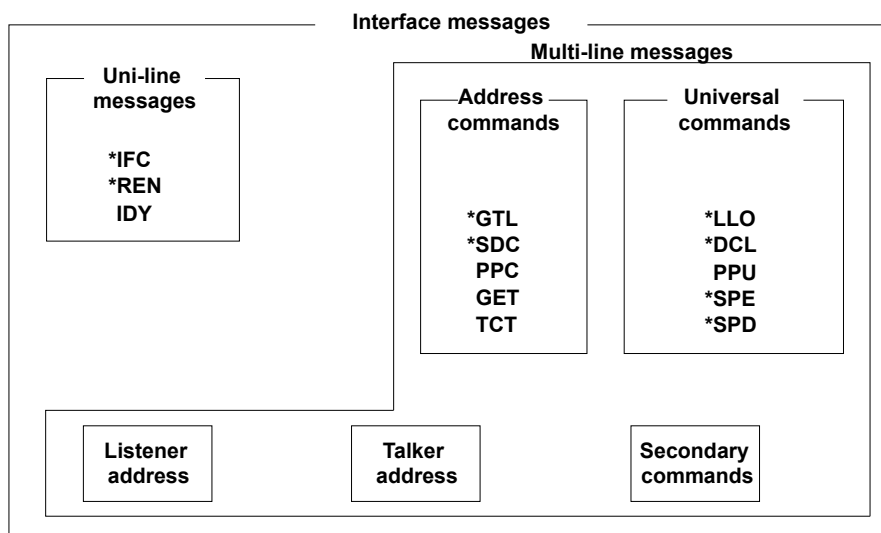
- TCT (Take Control)

- **Universal commands**

These commands are valid on all instruments regardless of the listener and talker designations. The following five types are available.

- LLO (Local Lockout)
- DCL (Device Clear)
- PPU (Parallel Poll Unconfigure)
- SPE (Serial Poll Enable)
- SPD (Serial Poll Disable)

In addition, listener address, talker address, and secondary commands are also considered interface messages.



Interface messages that DL9000 supports are indicated with \*marks.

### Note

The Differences between SDC and DCL

In multi-line messages, SDC messages are those that require talker or listener designation and DCL messages are those that do not require the designation. Therefore, SDC messages are directed at a particular instrument while DCL messages are directed at all instruments on the bus.

# 4.1 Messages

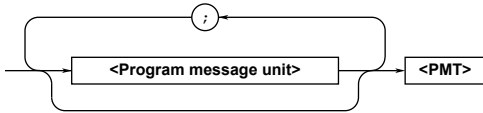
## Messages

Messages are used to exchange information between the controller and the instrument. Messages that are sent from the controller to the instrument are called program messages and messages that are sent back from the instrument to the controller are called response messages.

If a program message contains a message unit that requests a response (a query), the instrument returns a response message upon receiving the program message. A single response message is always returned in response to a single program message.

## Program Messages

The program message format is shown below



### <Program Message Unit>

A program message consists of one or more program message units; each unit corresponds to one command. The instrument executes the received commands in order.

Each program message unit is separated by a semicolon (;).

For details regarding the format of the program message unit, see the next section.

### Example

```
:ACQuire:MODE NORMAL;HRMode 1<PMT>
```

Unit                      Unit

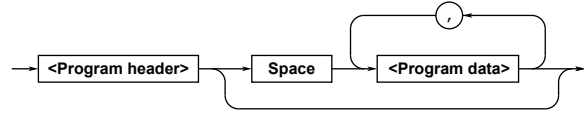
### <PMT>

PMT is a program message terminator. The following three types are available.

- NL (New Line): Same as LF (Line Feed). ASCII code "0AH"
- ^EOM: The END message as defined by USBTMC (The data byte that is sent simultaneously with the END message is the last data of the program message.)
- NL^EOM: NL with an END message added (NL is not included in the program message.)

## Program Message Unit Format

The program message unit format is shown below



### <Program Header>

The program header indicates the command type. For details, see page 4-3.

### <Program Data>

If certain conditions are required in executing a command, program data is added. A space (ASCII code "20H") separates the program data from the header. If there are multiple sets of program data, they are separated by commas (,).

For details, see page 4-5.

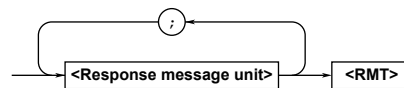
### Example

```
:ACQuire:MODE NORMAL<PMT>
```

Header                      Data

## Response Messages

The response message format is shown below.



### <Response Message Unit>

A response message consists of one or more response message units; each response message unit corresponds to one response.

Response message units are separated by a semicolon (;).

For details regarding the format of the response message unit, see the next section.

### Example

```
:ACQUIRE:MODE NORMAL;HRMode 1<RMT>
```

Unit                      Unit

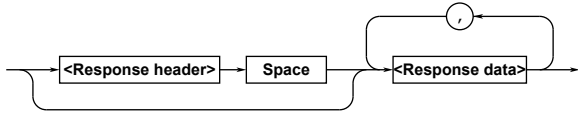
### <RMT>

A response message terminator. It is NL^EOM.

## 4.1 Messages

### Response Message Unit Format

The response message unit format is shown below.



#### <Response Header>

A response header sometimes precedes the response data. A space separates the data from the header. For details, see page 4-4.

#### <Response Data>

Response data contains the content of the response. If there are multiple sets of response data, they are separated by commas ( , ). For details, see page 4-5.

#### Example

$1.25E-02$ <RMT> :ACQUIRE:MODE NORMAL<RMT>  
Data Header Data

If there are multiple queries in a program message, responses are made in the same order as the queries. In most cases, a single query returns a single response message unit, but there are a few queries that return multiple units. The first response message unit always corresponds to the first query, but the  $n^{\text{th}}$  response unit may not necessarily correspond to the  $n^{\text{th}}$  query. Therefore, if you want to make sure that every response is retrieved, divide the program messages into individual messages.

### Precautions to Be Taken when Transferring Messages

- If a program message that does not contain a query is sent, the next program message can be sent at any time.
- If a program message that contains a query is sent, a response message must be received before the next program message can be sent. If the next program message is sent before the response message is received in its entirety, an error occurs. The response message that was not received is discarded.
- If the controller tries to receive a response message when there is none, an error occurs. If the controller tries to receive a response message before the transmission of the program message is complete, an error occurs.

- If a program message containing multiple message units is sent, and the message contains incomplete units, the instrument attempts to execute the ones that are believed to be complete. However, these attempts may not always be successful. In addition, if the message contains queries, the responses may not be returned.

### Deadlock

The instrument can store in its buffer program and response messages of length 1024 bytes or more (The number of available bytes varies depending on the operating conditions). When both the transmit and receive buffers become full at the same time, the instrument can no longer continue to operate. This state is called a deadlock. In this case, operation can be resumed by discarding the program message. Deadlock will not occur if the program message (including the <PMT>) is kept below 1024 bytes. Furthermore, deadlock never occurs if a program message does not contain a query.

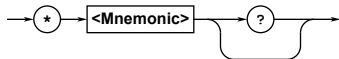
## 4.2 Commands

### Commands

There are three types of commands (program headers) that are sent from the controller to the instrument. They differ in their program header formats.

#### Common Command Header

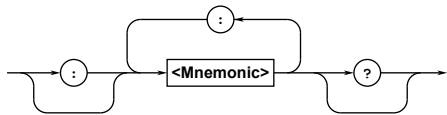
Commands that are defined in the USBTMC-USB488 are called common commands. The header format of a common command is shown below. An asterisk (\*) is always placed in the beginning of a command.



Common command example: \*CLS

#### Compound Header

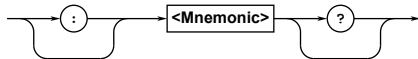
Dedicated commands used by the instrument are classified and arranged in a hierarchy according to their functions. The format of a compound header is shown below. A colon (:) must be used to specify a lower hierarchy.



Compound header example: :ACquire:MODE

#### Simple Header

These commands are functionally independent and do not have a hierarchy. The format of a simple header is shown below.



Simple header example: :START

#### Note

A <mnemonic> is a character string made up of alphanumeric characters.

### When Concatenating Commands

#### • Command Group

A command group is a group of commands that have common compound headers arranged in a hierarchy. A command group may contain sub-groups.

Example Group of commands related to acquisition

```
:ACquire:AVERage:COUNT
:ACquire:MODE
:ACquire:AVERage:EWEight
:ACquire:REPetitive
:ACquire:RLENgth
:ACquire:INTERLeave
```

#### • When Concatenating Commands of the Same Group

The instrument stores the hierarchical level of the command that is currently being executed, and performs analysis on the assumption that the next command sent will also belong to the same level. Therefore, common header sections can be omitted for commands belonging to the same group.

Example :ACquire:MODE NORMal;  
INTERLeave 1<PMT>

#### • When Concatenating Commands of Different Groups

If the following command does not belong to the same group, a colon (:) is placed in front of the header (cannot be omitted).

Example :ACquire:MODE NORMal;:DISPlay:  
FORMat SINGLE<PMT>

#### • When Concatenating Simple Headers

If a simple header follows another command, a colon (:) is placed in front of the simple header (cannot be omitted).

Example :ACquire:MODE NORMal;:  
START<PMT>

#### • When Concatenating Common Commands

Common commands that are defined in the USBTMC-USB488 are independent of hierarchy. Colons (:) are not needed before a common command.

Example :ACquire:MODE NORMal;\*CLS;  
INTERLeave 1<PMT>

#### • When Separating Commands with <PMT>

If a terminator is used to separate two commands, each command is a separate message. Therefore, the common header must be specified for each command even when commands belonging to the same command group are being concatenated.

Example :ACquire:MODE NORMal<PMT>:  
ACquire:INTERLeave 1<PMT>

## 4.2 Commands

---

### Upper-Level Query

An upper-level query is a query in which a question mark (?) is appended to the highest level command of a group. Execution of an upper-level query allows all settings that can be specified in the group to be received at once. Some query groups which are comprised of more than three hierarchical levels can output all the lower level settings.

**Example** :ACQUIRE? -> :ACQUIRE:AVERAGE:  
COUNT 2;EWEIGHT 16;:ACQUIRE:  
HRMODE 0;INTERLEAVE 0;  
INTERPOLATE 1;MODE NORMAL;  
REPETITIVE 0;RLENGTH 12500

The response to an upper-level query can be transmitted as a program message back to the instrument. In this way, the settings that existed when the upper-level query was made can be restored. However, some upper-level queries do not return setup information that is not currently in use. It is important to remember that not all the group's information is necessarily returned as part of a response.

### Header Interpretation Rules

The instrument interprets the header that is received according to the rules below.

- Mnemonics are not case sensitive.  
Example "CURSor" can also be written as "cursor"  
or "Cursor."
- The lower-case section of the header can be omitted.  
Example "CURSor" can also be written as "CURSO"  
or "CURS."
- The question mark (?) at the end of a header indicates that it is a query. The question mark (?) cannot be omitted.  
Example The shortest abbreviation for CURSor? is CURS?.
- If the <x> (value) at the end of a mnemonic is omitted, it is interpreted as a 1.  
Example If "CHANnel<x>" is written as "CHAN," it means "CHANnel1."
- **The section enclosed by braces ([]) can be omitted.**  
Example HISTory[:CURRent]:MODE can also be written as HIST:MODE.  
However, the last section enclosed by braces ([]) cannot be omitted in an upper-level query.  
Example "HISTory?" and "HISTory:CURRent?" are different queries.

## 4.3 Response

### Response

When the controller sends a message unit that has a question mark (?) in its program header (query), the instrument returns a response message to the query. A response message is returned in one of the following two forms.

- **Response Consisting of a Header and Data**  
If the response can be used as a program message without any change, it is returned with a command header attached.  
Example :ACQUire:MODE?<PMT>  
-> :ACQUire:MODE NORMAL<RMT>

- **Response Consisting of Data Only**  
If the response cannot be used as a program message unless changes are made to it (query-only command), only the data section is returned. However, there are query-only commands that return responses with the header attached.  
Example :MEASure:TRACel:AREAL:PTOPeak:VALue?<PMT> -> 10.0E+00<RMT>

### When You Wish to Return a Response without a Header

Responses that return both header and data can be set so that only the data section is returned. The "COMMunicate:HEADer" command is used to do this.

### Abbreviated Form

Normally, the lower-case section is removed from a response header before the response is returned to the controller. Naturally, the full form of the header can also be used. For this, the "COMMunicate:VERBose" command is used. The sections enclosed by braces ([]) are also omitted in the abbreviated form.

## 4.4 Data

### Data

A data section comes after the header. A space must be included between the header and the data. The data contains conditions and values. Data is classified as below.

Data	Meaning
<Decimal>	A value expressed as a decimal number (Example: Probe attenuation of CH1 -> CHANnel1:PROBe 100)
<Voltage><Time>	A physical value
<Frequency>	(Example: Time axis range)
<Current>	-> TIMEbase:TDIV 1US)
<Register>	Register value expressed as binary, octal, decimal or hexadecimal. (Example: Extended event register value -> STATUS:EeSE #HFE)
<Character Data>	Predefined character string (mnemonic). Can be selected from { }. (Example: Select the input coupling of CH1 -> CHANnel1:COUPling {AC DC DC50 GND})
<Boolean>	Indicates ON and OFF. Set using ON, OFF or a value (Example: Turn ON the CH1 display -> CHANnel1:DISPlay ON)
<String data>	An arbitrary character string (Example: Comment to a screen data output -> MATH1:UNIT:USERdefine "VOLT")
<Filename>	Indicates a file name. (Example: Save file name -> FILE:SAVE:WAVEform:NAME "CASE1")
<Block data>	Arbitrary 8-bit data (Example: Response to acquired waveform data -> #800000010ABCDEFGHIJ)

### <Decimal>

<Decimal> indicates a value expressed as a decimal number, as shown in the table below. Decimal values are given in the NR form as specified in the ANSI X3.42-1975.

Symbol	Meaning	Example
<NR1>	Integer	125-1 +1000
<NR2>	Fixed-point number	125.0 -.90 +001.
<NR3>	Floating-point number	125.0E+0 -9E-1 +.1E4
<NRf>	Any of the forms <NR1> to <NR3> is allowed.	

- The instrument can receive decimal values that are sent from the controller in any of the forms, <NR1> to <NR3>. This is represented by <NRf>.
- For response messages that the instrument returns to the controller, the form (<NR1> to <NR3> to be used) is determined by the query. The same form is used regardless of the size of the value.
- For the <NR3> format, the "+" sign after the "E" can be omitted. However, the "-" sign cannot be omitted.
- If a value outside the setting range is entered, the value is normalized so that it is just inside the range.
- If a value has more significant digits than the available resolution, the value is rounded.

### <Voltage>, <Time>, <Frequency>, and <Current>

<Voltage>, <Time>, <Frequency>, and <Current> indicate decimal values that have physical significance. <Multiplier> or <Unit> can be attached to the <NRf> form that was described earlier. It is expressed in one of the following forms.

Form	Example
<NRf><Multiplier><Unit>	5MV
<NRf><Unit>	5E-3V
<NRf><Multiplier>	5M
<NRf>	5E-3

### <Multiplier>

<Multipliers> which can be used are indicated below.

Symbol	Word	Multiplier
EX	Exa	10 <sup>18</sup>
PE	Peta	10 <sup>15</sup>
T	Tera	10 <sup>12</sup>
G	Giga	10 <sup>9</sup>
MA	Mega	10 <sup>6</sup>
K	Kilo	10 <sup>3</sup>
M	Milli	10 <sup>-3</sup>
U	Micro	10 <sup>-6</sup>
N	Nano	10 <sup>-9</sup>
P	Pico	10 <sup>-12</sup>
F	Femto	10 <sup>-15</sup>
A	Ato	10 <sup>-18</sup>

### <Unit>

<Units> that can be used are indicated below.

Symbol	Word	Meaning
V	Volt	Voltage
S	Second	Time
HZ	Hertz	Frequency
MHZ	Megahertz	Frequency
A	Ampere	Current

- <Multiplier> and <Unit> are not case sensitive.
- "U" is used to indicate micro "μ".
- "MA" is used for Mega to distinguish it from Milli. The only exception is Megahertz which is expressed as "MHZ." Therefore, the "M (Milli)" multiplier cannot be used for frequencies.
- If both <Multiplier> and <Unit> are omitted, the default unit is used.
- Response messages are always expressed in the <NR3> form. Response messages are returned using the default unit without the <Multiplier> or <Unit>.

**<Register>**

<Register> indicates an integer, and can be expressed in hexadecimal, octal, or binary as well as a decimal number. It is used when each bit of the value has a particular meaning. It is expressed in one of the following forms.

Form	Example
<NRf>	1
#H<Hexadecimal value made up of the digits 0 to 9 and A to F>	#H0F
#Q<Octal value made up of the digits 0 to 7>	#Q777
#B<Binary value made up of the digits 0 and 1>	#B001100

- <Register> is not case sensitive.
- Response messages are always expressed as <NR1>.

**<Character Data>**

<Character Data> is a specified string of character data (a mnemonic). It is mainly used to indicate options and is chosen from the character strings given in { }. For interpretation rules, refer to "Header Interpretation Rules" on page 4-4.

Form	Example
{AC DC DC50 GND}	AC

- As with the header, the "COMMunicate:VERBoSe" command can be used to select whether to return the response in the full form or in the abbreviated form.
- The "COMMunicate:HEADer" setting does not affect the character data.

**<Boolean>**

<Boolean> is data that indicates ON or OFF. It is expressed in one of the following forms.

Form	Example
{ON OFF <NRf>}	ON    OFF    1    0

- When <Boolean> is expressed in the <NRf> form, "OFF" is selected if the rounded integer value is 0, and ON for all other cases.
- A response message is always returned with a 1 if the value is ON and 0 if the value is OFF.

**<String data>**

<String data> is not a specified character string like <Character data>. It is an arbitrary character string. The character string must be enclosed in single quotation marks (') or double quotation marks (").

Form	Example
<String data>	'ABC'    "IEEE488.2-1987"

- If a character string contains a double quotation mark ("), the double quotation mark is replaced by two double quotation marks (""). This rule also applies to a single quotation mark within a character string.
- A response message is always enclosed in double quotation marks (").
- <String data> is an arbitrary character string. Therefore the instrument assumes that the remaining program message units are part of the character string if no single (') or double quotation mark (") is encountered. As a result, no error is detected if a quotation mark is omitted.

**<Filename>**

<Filename> is data that indicates a file name. It is expressed in one of the following forms.

Form	Example
{<NRf> <Character data> <String data>}	1    CASE    "CASE"

- <NRf> is rounded to an 8-digit integer and converted to ASCII code. The result is the file name (example: 1 becomes "00000001"). Negative values are not allowed.
- Response messages are always returned in the <String data> form.
- For <Character data>, the first 12 characters become the file name.
- For <String data>, the first 259 characters become the file name.
- For a description of the number of characters of the <String data> file name, see the *DL9000 User's Manual*.

**<Block data>**

<Block data> is arbitrary 8-bit data. It is only used in response messages on the DL9000. Below is the syntax.

Form	Example
#N<N-digit decimal number>	#800000010ABCDEFGHIJ
<Data byte sequence>	

- #N  
Indicates that the data is <Block data>. "N" indicates the number of succeeding data bytes (digits) in ASCII code characters.
- <N-digit decimal number>  
Indicates the number of bytes of data (example: 00000010 = 10 bytes).
- <data byte sequence>  
Expresses the actual data (example: ABCDEFGHIJ).
- Data is comprised of 8-bit values (0 to 255). This means that the ASCII code "0AH," which stands for "NL," can also be a code used for data. Hence, care must be taken when programming the controller.



## 4.5 Synchronization with the Controller

### Overlap Commands and Sequential Commands

There are two types of commands, overlap commands and sequential commands. In the case of overlap commands, the execution of the next command may start before the execution of the previous command is completed.

For example, if the next program message is transmitted when specifying the V/div value and querying the result, the response always returns the most recent setting (5 V in this case).

```
:CHANnel1:VDIV 5V;VDIV?<PMT>
```

This is because the next command is forced to wait until the processing of "CHANnel1:VDIV" itself is completed. This type of command is called a sequential command.

On the contrary, let us assume that you send the next program message when you wish to load a file and query the V/div value of the result.

```
:FILE:LOAD:SETup:EXECute "CASE1";:  
CHANnel1:VDIV?
```

In this case, "CHANnel1:VDIV?" is executed before the loading of the file is completed, and the V/div value that is returned is the value before the file is loaded. The act of executing the next command before the processing of itself is completed such as with "FILE:LOAD:SETup:EXECute "CASE1"" is called an overlap operation. A command that operates in this way is called an overlap command.

In such case, the overlap operation can be prevented by using the methods below.

### Synchronizing with Overlap Commands

#### • Using the \*WAI Command

The \*WAI command holds the subsequent commands until the overlap command is completed.

```
Example :COMMunicate:OPSE #H0040;:  
FILE:LOAD:SETup:  
EXECute "CASE1";*WAI;:  
CHANnel1:VDIV?<PMT>
```

"COMMunicate:OPSE" is a command used to select the "\*WAI" target. Here, media access is specified.

Because "\*WAI" is executed immediately before "CHANnel1:VDIV?," "CHANnel1:VDIV?" is not executed until the file loading is complete.

#### • Using the COMMunicate:OVERlap command

The COMMunicate:OVERlap command enables (or disables) overlap operation.

```
Example :COMMunicate:OVERlap #HFFBF;:  
FILE:LOAD:SETup:  
EXECute "CASE1";:CHANnel1:  
VDIV?<PMT>
```

"COMMunicate:OVERlap #HFFBF" enables overlap operation on commands other than media access. Because the overlap operation of file loading is disabled, "FILE:LOAD:SETup:EXECute "CASE1"" operates in the same way as a sequential command. Therefore, CHANnel1:VDIV? is not executed until the file loading is complete.

#### • Using the \*OPC Command

The \*OPC command sets the OPC bit, bit 0 of the standard event register (see page 6-3), to 1 when the overlap operation is completed.

```
Example :COMMunicate:OPSE #H0040;  
*ESE 1;  
*ESR?;*SRE 32;:FILE:LOAD:SETup:  
EXECute "CASE1";*OPC<PMT>  
(Read the response to *ESR?)  
(Wait for a service request)  
:CHANnel1:VDIV?<PMT>
```

"COMMunicate:OPSE" is a command used to select the "\*OPC" target. Here, media access is specified. "\*ESE 1" and "\*SRE 32" indicate that a service request is generated only when the OPC bit is 1. "\*ESR?" clears the standard event register.

In the example above, "CHANnel1:VDIV?" is not executed until a service request is generated.

- **Using the \*OPC? Query**

The \*OPC? query generates a response when an overlap operation is completed.

```
Example :COMMunicate:OPSE #H0040;:FILE:
LOAD:SETup:EXECute "CASE1";
*OPC?<PMT>
(Read the response to *OPC?)
:CHANnel1:VDIV?<PMT>
```

“COMMunicate:OPSE” is a command used to select the “\*OPC?” target. Here, media access is specified.

Because “\*OPC?” does not generate a response until the overlap operation is completed, the loading of the file will have been completed by the time the response to “\*OPC?” is read.

---

**Note**

Most commands are sequential commands. Overlap commands are indicated as overlap commands in chapter 5. All other commands are sequential commands.

---

### Achieving Synchronization without Using Overlap Commands

Even for sequential commands, synchronization is sometimes required for non communication-related reasons such as a trigger occurrence.

For example, if the next program message is transmitted to make an inquiry about the waveform data which has been acquired with the trigger mode set to single, the WAVEform:SEND? command may be executed regardless of whether the acquisition has been completed or not and may result in command execution error.

```
TRIGger:MODE SINGLE;:START;:WAVEform:
SEND?<PMT>
```

In this case, the following method must be used to synchronize with the end of the acquisition.

- **Using the STATus:CONDition? Query**

The “STATus:CONDition?” query is used to query the contents of the condition register (page 6-4).

Whether waveforms are being retrieved can be determined by reading bit 0 of the condition register. If bit 0 of the condition register is “1,” waveforms are being retrieved. Otherwise, it is stopped.

```
Example TRIGger:MODE SINGLE;:START<PMT>
:STATus:CONDition?<PMT>
(Read the response. If bit 0 is 1, repeat
this command until it becomes 1.)
:WAVEform:SEND?<PMT>
```

The WAVEform:SEND? command will not be executed until bit 0 of the condition register is set to “0.”

- **Using the Extended Event Register**

The changes in the condition register can be reflected in the extended event register (page 6-4).

```
Example :STATus:FILTer1 FALL;:
STATus:EESR 1;EESR?;*SRE 8;:
TRIGger:MODE SINGLE;:START<PMT>
(Read the response to STATus:EESR?)
(Wait for a service request)
:WAVEform:SEND?<PMT>
```

The “STATus:FILTer1 FALL” command sets the transition filter so that bit 0 (FILTer1) of the extended event register is set to 1 when bit 0 of the condition register changes from 1 to 0.

The “:STATus:EESR 1” command is used to reflect only bit 0 of the extended event register to the status byte.

The “STATus:EESR?” command is used to clear the extended event register.

The “\*SRE 8” command is used to generate a service request solely on the cause of the extended event register.

The “WAVEform:SEND?” command is not executed until a service request is generated.

- **Using the COMMunicate:WAIT Command**

The “COMMunicate:WAIT” command halts communications until a specific event is generated.

```
Example :STATus:FILTer1 FALL;:
STATus:EESR?;:TRIGger:
MODE SINGLE<PMT>
(Read the response to STATus:EESR?)
:COMMunicate:WAIT 1;:WAVEform:
SEND?<PMT>
```

For a description of “STATus:FILTer1 FALL” and “STATus:EESR?” see the previous section regarding the extended event register.

The “COMMunicate:WAIT 1” command indicates that the program will wait for bit 0 of the extended event register to be set to “1.”

The WAVEform:SEND? command will not be executed until bit 0 of the extended event register is set to “1.”

## 5.1 List of Commands

Command	Function	Page
<b>ACQUIRE Group</b>		
:ACQUIRE?	Queries all settings related to the waveform acquisition.	5-34
:ACQUIRE:AVERage?	Queries all settings related to averaging and the waveform acquisition count.	5-34
:ACQUIRE:AVERage:COUNT	Sets the waveform acquisition count of averaging mode or queries the current setting.	5-34
:ACQUIRE:AVERage:EWEight	Sets the attenuation constant when averaging mode is used infinitely or queries the current setting.	5-34
:ACQUIRE:HRMode	Turns ON/OFF the high resolution mode or queries the current setting.	5-34
:ACQUIRE:INTERLeave	Turns ON/OFF interleave or queries the current setting.	5-34
:ACQUIRE:INTERPolate	Turns ON/OFF data interpolation or queries the current setting.	5-34
:ACQUIRE:MODE	Sets the waveform acquisition mode or queries the current setting.	5-34
:ACQUIRE:REPetitive	Turns ON/OFF the repetitive sampling or queries the current setting.	5-34
:ACQUIRE:RENgth	Sets the record length or queries the current setting.	5-34
<b>ANALYSIS Group</b>		
:ANALYSIS?	Queries all settings related to the analysis function.	5-35
:ANALYSIS:AHIStogram<x>?	Queries all settings related to the accumulated histogram function.	5-35
:ANALYSIS:AHIStogram<x>: HORizontal	Sets the horizontal range of the accumulated histogram or queries the current setting.	5-35
:ANALYSIS:AHIStogram<x>: MEASure?	Queries all settings related automated measurement of the accumulated histogram.	5-36
:ANALYSIS:AHIStogram<x>: MEASure:CURSor?	Queries all settings related to cursor measurement of the accumulated histogram.	5-36
:ANALYSIS:AHIStogram<x>: MEASure:CURSor:BASic?	Queries all settings related to basic items of the cursor of the accumulated histogram.	5-36
:ANALYSIS:AHIStogram<x>: MEASure:CURSor[:BASic]:ALL	Turns ON/OFF all basic items of the cursor of the accumulated histogram.	5-36
:ANALYSIS:AHIStogram<x>: MEASure:CURSor[:BASic]:C<x>?	Queries all settings related to the cursor of the accumulated histogram.	5-37
:ANALYSIS:AHIStogram<x>: MEASure:CURSor[:BASic]:C<x>: STATe	Turns ON/OFF the cursor of the accumulated histogram or queries the current setting.	5-37
:ANALYSIS:AHIStogram<x>: MEASure:CURSor[:BASic]:C<x>: VALue?	Queries the measured value of the cursor of the accumulated histogram.	5-37
:ANALYSIS:AHIStogram<x>: MEASure:CURSor[:BASic]:DC?	Queries all settings related to measured values between cursors of the accumulated histogram.	5-37
:ANALYSIS:AHIStogram<x>: MEASure:CURSor[:BASic]:DC:STATe	Turns ON/OFF the measured values between cursors of the accumulated histogram or queries the current setting.	5-37
:ANALYSIS:AHIStogram<x>: MEASure:CURSor[:BASic]:DC: VALue?	Queries the measured value between cursors of the accumulated histogram.	5-37
:ANALYSIS:AHIStogram<x>: MEASure:CURSor:CALCulation?	Queries all settings related to calculation items of the cursor of the accumulated histogram.	5-37
:ANALYSIS:AHIStogram<x>: MEASure:CURSor:CALCulation:ALL	Turns ON/OFF all calculation items of the cursor of the accumulated histogram.	5-37
:ANALYSIS:AHIStogram<x>: MEASure:CURSor:CALCulation: DEFine<x>	Sets the equation of the calculation item of the cursor of the accumulated histogram or queries the current setting.	5-38
:ANALYSIS:AHIStogram<x>: MEASure:CURSor:CALCulation: STATe<x>	Turns ON/OFF the calculation item of the cursor of the accumulated histogram or queries the current setting.	5-38
:ANALYSIS:AHIStogram<x>: MEASure:CURSor:CALCulation: VALue<x>?	Queries the measured value of the calculation item of the cursor of the accumulated histogram.	5-38
:ANALYSIS:AHIStogram<x>: MEASure:CURSor:HLINKage	Turns ON/OFF the horizontal cursor link of the accumulated histogram or queries the current setting.	5-38

## 5.1 List of Commands

Command	Function	Page
:ANALysis:AHIStogram<x>: MEASure:CURSor:HPOSition<x>	Sets the horizontal cursor position of the accumulated histogram or queries the current setting.	5-38
:ANALysis:AHIStogram<x>: MEASure:CURSor:VLINKage	Turns ON/OFF the vertical cursor link of the accumulated histogram or queries the current setting.	5-38
:ANALysis:AHIStogram<x>: MEASure:CURSor:VPOSition<x>	Sets the vertical cursor position of the accumulated histogram or queries the current setting.	5-39
:ANALysis:AHIStogram<x>: MEASure:MODE	Sets the automated measurement mode of the accumulated histogram or queries the current setting.	5-39
:ANALysis:AHIStogram<x>: MEASure:PARAmeter?	Queries all settings related to the automated measurement of waveform parameters of the accumulated histogram.	5-39
:ANALysis:AHIStogram<x>: MEASure:PARAmeter:AREA<x>?	Queries all settings related to the area of the accumulated histogram.	5-40
:ANALysis:AHIStogram<x>: MEASure:PARAmeter:AREA<x>:ALL	Turns ON/OFF all waveform parameters of the accumulated histogram.	5-40
:ANALysis:AHIStogram<x>: MEASure:PARAmeter: AREA<x>:<Parameter>?	Queries all settings related to the waveform parameter of the accumulated histogram.	5-40
:ANALysis:AHIStogram<x>: MEASure:PARAmeter: AREA<x>:<Parameter>:STATe	Turns ON/OFF the waveform parameter of the accumulated histogram or queries the current setting.	5-40
:ANALysis:AHIStogram<x>: MEASure:PARAmeter: AREA<x>:<Parameter>:VALue?	Queries the automated measured value of the waveform parameter of the accumulated histogram.	5-41
:ANALysis:AHIStogram<x>: MEASure:PARAmeter:CALCulation?	Queries all settings related to the calculation items of waveform parameters of the accumulated histogram.	5-41
:ANALysis:AHIStogram<x>: MEASure:PARAmeter:CALCulation: ALL	Turns ON/OFF all calculation items of the waveform parameters of the accumulated histogram.	5-41
:ANALysis:AHIStogram<x>: MEASure:PARAmeter:CALCulation: DEFine<x>	Sets the equation of the calculation items of the waveform parameter of the accumulated histogram or queries the current setting.	5-41
:ANALysis:AHIStogram<x>: MEASure:PARAmeter:CALCulation: STATe<x>	Turns ON/OFF the calculation items of the waveform parameter of the accumulated histogram or queries the current setting.	5-41
:ANALysis:AHIStogram<x>: MEASure:PARAmeter:CALCulation: VALue<x>?	Queries the automated measured value of the calculation items of the waveform parameter of the accumulated histogram.	5-41
:ANALysis:AHIStogram<x>: MEASure:PARAmeter:HRANge<x>	Sets the horizontal range of the waveform parameter of the accumulated histogram or queries the current setting.	5-42
:ANALysis:AHIStogram<x>: MEASure:PARAmeter:VRANge<x>	Sets the vertical range of the waveform parameter of the accumulated histogram or queries the current setting.	5-42
:ANALysis:AHIStogram<x>:MODE	Sets the accumulated histogram mode or queries the current setting.	5-42
:ANALysis:AHIStogram<x>:TRACe	Sets the source trace of the accumulated histogram or queries the current setting.	5-42
:ANALysis:AHIStogram<x>: VERTical	Sets the vertical range of the accumulated histogram or queries the current setting.	5-42
:ANALysis:AHIStogram<x>:WINDow	Sets the measurement target window of the accumulated histogram or queries the current setting.	5-42
:ANALysis:DISPlay<x>	Turns ON/OFF the analysis function display or queries the current setting.	5-42
:ANALysis:FFT<x>?	Queries all settings related to the FFT computation function.	5-43
:ANALysis:FFT<x>:HORizontal?	Queries all settings related the horizontal axis of the FFT computation.	5-43
:ANALysis:FFT<x>:HORizontal: CSPan?	Queries all settings related to the center and span of the horizontal axis of the FFT computation.	5-43
:ANALysis:FFT<x>:HORizontal: CSPan:CENTer	Sets the horizontal center of the FFT computation or queries the current setting.	5-43
:ANALysis:FFT<x>:HORizontal: CSPan:SPAN	Sets the horizontal span of the FFT computation or queries the current setting.	5-43
:ANALysis:FFT<x>:HORizontal: LRIGHT?	Queries all settings related the left and right edges of the horizontal axis of the FFT computation.	5-43
:ANALysis:FFT<x>:HORizontal: LRIGHT:RANGe	Sets the range of the horizontal left and right edges of the FFT computation or queries the current setting.	5-44
:ANALysis:FFT<x>:HORizontal: MODE	Sets the horizontal mode of the FFT computation or queries the current setting.	5-44
:ANALysis:FFT<x>:IPART (Imag Part)	Sets the source trace of the imaginary part of the FFT computation or queries the current setting.	5-44

## 5.1 List of Commands

Command	Function	Page
:ANALysis:FFT<x>:LENGth	Sets the number of FFT points of the FFT computation or queries the current setting.	5-44
:ANALysis:FFT<x>:MAXHold	Turns ON/OFF the maximum value hold function of the FFT computation or queries the current setting.	5-44
:ANALysis:FFT<x>:MEASure?	Queries all settings related to the automated measurement of the FFT computation.	5-44
:ANALysis:FFT<x>:MEASure:MARKer?	Queries all settings related to the marker cursor measurement of the FFT computation.	5-45
:ANALysis:FFT<x>:MEASure:MARKer:BASic?	Queries all settings related to basic items of the marker cursor of the FFT computation.	5-45
:ANALysis:FFT<x>:MEASure:MARKer[:BASic]:ALL	Turns ON/OFF all basic items of the marker cursor of the FFT computation.	5-45
:ANALysis:FFT<x>:MEASure:MARKer[:BASic]:DFRequency?	Queries all settings related to the frequency value between marker cursors of the FFT computation.	5-45
:ANALysis:FFT<x>:MEASure:MARKer[:BASic]:DFRequency:STATe	Turns ON/OFF the frequency value between marker cursors of the FFT computation or queries the current setting.	5-45
:ANALysis:FFT<x>:MEASure:MARKer[:BASic]:DFRequency:VALue?	Queries the frequency value between marker cursors of the FFT computation.	5-45
:ANALysis:FFT<x>:MEASure:MARKer[:BASic]:DV?	Queries all settings related to the power value between marker cursors of the FFT computation.	5-46
:ANALysis:FFT<x>:MEASure:MARKer[:BASic]:DV:STATe	Turns ON/OFF the power value between marker cursors of the FFT computation or queries the current setting.	5-46
:ANALysis:FFT<x>:MEASure:MARKer[:BASic]:DV:VALue?	Queries the power value between marker cursors of the FFT computation.	5-46
:ANALysis:FFT<x>:MEASure:MARKer[:BASic]:FREquency<x>?	Queries all settings related to the frequency value of the marker cursor of the FFT computation.	5-46
:ANALysis:FFT<x>:MEASure:MARKer[:BASic]:FREquency<x>:STATe	Turns ON/OFF the frequency value of the marker cursor of the FFT computation or queries the current setting.	5-46
:ANALysis:FFT<x>:MEASure:MARKer[:BASic]:FREquency<x>:VALue?	Queries the frequency value of the marker cursor of the FFT computation.	5-46
:ANALysis:FFT<x>:MEASure:MARKer[:BASic]:POSition<x>{<Nrf>}	Sets the marker cursor position of the FFT computation or queries the current setting.	5-46
:ANALysis:FFT<x>:MEASure:MARKer[:BASic]:V<x>?	Queries all settings related to the power value of the marker cursor of the FFT computation.	5-47
:ANALysis:FFT<x>:MEASure:MARKer[:BASic]:V<x>:STATe	Turns ON/OFF the power value of the marker cursor of the FFT computation or queries the current setting.	5-47
:ANALysis:FFT<x>:MEASure:MARKer[:BASic]:V<x>:VALue?	Queries the power value of the marker cursor of the FFT computation.	5-47
:ANALysis:FFT<x>:MEASure:MARKer:CALCulation?	Queries all settings related to calculation items of the marker cursor of the FFT computation.	5-47
:ANALysis:FFT<x>:MEASure:MARKer:CALCulation:ALL	Turns ON/OFF all calculation items of the marker cursor of the FFT computation.	5-47
:ANALysis:FFT<x>:MEASure:MARKer:CALCulation:DEFine<x>	Sets the equation of the calculation items of the marker cursor of the FFT computation or queries the current setting.	5-47
:ANALysis:FFT<x>:MEASure:MARKer:CALCulation:STATe<x>	Turns ON/OFF the calculation items of the marker cursor of the FFT computation or queries the current setting.	5-47
:ANALysis:FFT<x>:MEASure:MARKer:CALCulation:VALue<x>?	Queries the measured value of the calculation items of the marker cursor of the FFT computation.	5-48
:ANALysis:FFT<x>:MEASure:MODE	Sets the automated measurement mode of the FFT computation or queries the current setting.	5-48
:ANALysis:FFT<x>:MEASure:PEAK?	Queries all settings related to the peak value measurement of the FFT computation.	5-48
:ANALysis:FFT<x>:MEASure:PEAK:BASic?	Queries all settings related to basic items of the peak value of the FFT computation.	5-48
:ANALysis:FFT<x>:MEASure:PEAK[:BASic]:ALL	Turns ON/OFF all basic items of the peak value of the FFT computation.	5-48
:ANALysis:FFT<x>:MEASure:PEAK[:BASic]:DFRequency?	Queries all settings related to the frequency value between peak values of the FFT computation.	5-48
:ANALysis:FFT<x>:MEASure:PEAK[:BASic]:DFRequency:STATe	Turns ON/OFF the frequency value between peak values of the FFT computation or queries the current setting.	5-49

## 5.1 List of Commands

Command	Function	Page
:ANALysis:FFT<x>:MEASure:PEAK[:BASic]:DFRequency:VALue?	Queries the frequency value between peak values of the FFT computation.	5-49
:ANALysis:FFT<x>:MEASure:PEAK[:BASic]:DV?	Queries all settings related to the power value between peak values of the FFT computation.	5-49
:ANALysis:FFT<x>:MEASure:PEAK[:BASic]:DV:STATe	Turns ON/OFF the power value between peak values of the FFT computation or queries the current setting.	5-49
:ANALysis:FFT<x>:MEASure:PEAK[:BASic]:DV:VALue?	Queries the power value between peak values of the FFT computation.	5-49
:ANALysis:FFT<x>:MEASure:PEAK[:BASic]:FREQuency<x>?	Queries all settings related to the peak frequency value of the FFT computation.	5-49
:ANALysis:FFT<x>:MEASure:PEAK[:BASic]:FREQuency<x>:STATe	Turns ON/OFF the peak frequency value of the FFT computation or queries the current setting.	5-49
:ANALysis:FFT<x>:MEASure:PEAK[:BASic]:FREQuency<x>:VALue?	Queries the peak frequency value of the FFT computation.	5-49
:ANALysis:FFT<x>:MEASure:PEAK[:BASic]:RANGe<x>	Sets the measurement range of the peak value of the FFT computation or queries the current setting.	5-50
:ANALysis:FFT<x>:MEASure:PEAK[:BASic]:V<x>?	Queries all settings related to the peak value of the FFT computation.	5-50
:ANALysis:FFT<x>:MEASure:PEAK[:BASic]:V<x>:STATe	Turns ON/OFF the peak value of the FFT computation or queries the current setting.	5-50
:ANALysis:FFT<x>:MEASure:PEAK[:BASic]:V<x>:VALue?	Queries the peak value of the FFT computation.	5-50
:ANALysis:FFT<x>:MEASure:PEAK:CALCulation?	Queries all settings related to calculation items of the FFT computation.	5-50
:ANALysis:FFT<x>:MEASure:PEAK:CALCulation:ALL	Turns ON/OFF all calculation items of the FFT computation.	5-50
:ANALysis:FFT<x>:MEASure:PEAK:CALCulation:DEFine<x>	Sets the equation of the calculation item of the FFT computation or queries the current setting.	5-50
:ANALysis:FFT<x>:MEASure:PEAK:CALCulation:STATe<x>	Turns ON/OFF the calculation items of the FFT computation or queries the current setting.	5-51
:ANALysis:FFT<x>:MEASure:PEAK:CALCulation:VALue<x>?	Queries the measured value of the calculation item of the FFT computation.	5-51
:ANALysis:FFT<x>:RANGe	Sets the measurement source window used in the FFT computation or queries the current setting.	5-51
:ANALysis:FFT<x>:RPART (Real Part)	Sets the source trace of the real part of the FFT computation or queries the current setting.	5-51
:ANALysis:FFT<x>:RPOSition (Ref Position)	Sets the center point of magnification of the vertical axis of the FFT computation or queries the current setting.	5-51
:ANALysis:FFT<x>:VERTical?	Queries all settings related to the vertical axis of the FFT computation.	5-51
:ANALysis:FFT<x>:VERTical:LEVel	Sets the display position of the vertical axis of the FFT computation or queries the current setting.	5-51
:ANALysis:FFT<x>:VERTical:MODE	Sets the vertical axis mode of the FFT computation or queries the current setting.	5-51
:ANALysis:FFT<x>:VERTical:SENSitivity	Sets the vertical sensitivity of the FFT computation or queries the current setting.	5-52
:ANALysis:FFT<x>:WINDow	Sets the window function of the FFT computation or queries the current setting.	5-52
:ANALysis:TYPE<x>	Sets the analysis function type or queries the current setting.	5-52
:ANALysis:VTDisplay	Turns ON/OFF the VT waveform display or queries the current setting.	5-52
:ANALysis:WAIT<x>?	Waits for the completion of the automated measurement with a timeout option.	5-52
:ANALysis:WPARAMeter<x>?	Queries all settings related to the waveform parameter measurement function.	5-52
:ANALysis:WPARAMeter<x>:CALCulation	Sets the calculation items of the automated measurement of waveform parameters or queries the current setting.	5-53
:ANALysis:WPARAMeter<x>:HISTogram?	Queries all settings related to the histogram display of the waveform parameter measurement.	5-53
:ANALysis:WPARAMeter<x>:HISTogram:MEASure?	Queries all settings related to the automated measurement of the histogram display of the waveform parameter measurement.	5-53
:ANALysis:WPARAMeter<x>:HISTogram:MEASure:CURSor?	Queries all settings related to the cursor measurement on the histogram of the waveform parameter measurement.	5-53
:ANALysis:WPARAMeter<x>:HISTogram:MEASure:CURSor:ALL	Turns ON/OFF all histogram cursors of waveform parameter measurement.	5-54

## 5.1 List of Commands

Command	Function	Page
:ANALysis:WPARameter<x>: HISTogram:MEASure:CURSor:C<x>?	Queries all settings related to the cursor measurement on the histogram of the waveform parameter measurement.	5-54
:ANALysis:WPARameter<x>: HISTogram:MEASure:CURSor:C<x>: POSition	Sets the cursor position on the trend of the waveform parameter measurement or queries the current setting.	5-54
:ANALysis:WPARameter<x>: HISTogram:MEASure:CURSor:C<x>: STATe	Turns ON/OFF the cursor on the histogram of the waveform parameter measurement or queries the current setting.	5-54
:ANALysis:WPARameter<x>: HISTogram:MEASure:CURSor:C<x>: VALue?	Queries the measured value of the cursor on the histogram of the waveform parameter measurement.	5-54
:ANALysis:WPARameter<x>: HISTogram:MEASure:CURSor:DC?	Queries all settings related to the measurement between cursors on the histogram of the waveform parameter measurement.	5-54
:ANALysis:WPARameter<x>: HISTogram:MEASure:CURSor:DC: STATe	Turns ON/OFF the measurement between cursors on the histogram of the waveform parameter measurement or queries the current setting.	5-55
:ANALysis:WPARameter<x>: HISTogram:MEASure:CURSor:DC: VALue?	Queries the measured value between cursors on the histogram of the waveform parameter measurement.	5-55
:ANALysis:WPARameter<x>: HISTogram:MEASure:CURSor: LINKage	Turns ON/OFF the histogram cursor link of the waveform parameter measurement or queries the current setting.	5-55
:ANALysis:WPARameter<x>: HISTogram:MEASure:MODE	Sets the automated measurement mode of the histogram display of the waveform parameter measurement or queries the current setting.	5-55
:ANALysis:WPARameter<x>: HISTogram:MEASure:PARAmeter?	Queries all settings related to the automated measurement of histogram parameters of the waveform parameter measurement.	5-55
:ANALysis:WPARameter<x>: HISTogram:MEASure:PARAmeter:ALL	Turn ON/OFF all histogram parameters of the waveform parameter measurement.	5-55
:ANALysis:WPARameter<x>: HISTogram:MEASure: PARAmeter:<Parameter>?	Queries all settings related to the histogram parameter of the waveform parameter measurement.	5-56
:ANALysis:WPARameter<x>: HISTogram:MEASure: PARAmeter:<Parameter>:STATe	Turns ON/OFF the histogram parameter of the waveform parameter measurement or queries the current setting.	5-56
:ANALysis:WPARameter<x>: HISTogram:MEASure: PARAmeter:<Parameter>:VALue?	Queries the measured value of the histogram parameter of the waveform parameter measurement.	5-56
:ANALysis:WPARameter<x>:LIST?	Queries all settings related to the list display of the waveform parameter measurement.	5-56
:ANALysis:WPARameter<x>:LIST: ITEM?	Queries list display items of the waveform parameter measurement.	5-56
:ANALysis:WPARameter<x>:LIST: MODE	Sets the list display mode of the waveform parameter measurement or queries the current setting.	5-56
:ANALysis:WPARameter<x>:LIST: SCRoll	Sets the scroll direction of the list display of the waveform parameter measurement or queries the current setting.	5-56
:ANALysis:WPARameter<x>:LIST: VALue?	Queries the automated measured value of the list display number of the waveform parameter measurement.	5-57
:ANALysis:WPARameter<x>:MODE	Sets the mode of the waveform parameter measurement or queries the current setting.	5-57
:ANALysis:WPARameter<x>: TRACe<x>?	Queries all settings related to the trace of the waveform parameter measurement.	5-57
:ANALysis:WPARameter<x>: TRACe<x>:AREA<x>?	Queries all settings related to the area of the waveform parameter measurement.	5-57
:ANALysis:WPARameter<x>: TRACe<x>:AREA<x>:TYPE	Sets the waveform parameter of the waveform parameter measurement or queries the current setting.	5-57
:ANALysis:WPARameter<x>:TRENd?	Queries all settings related to the trend display of the waveform parameter measurement.	5-57
:ANALysis:WPARameter<x>:TRENd: ASCale[:EXECute]	Executes the auto scaling of the trend display of the waveform parameter measurement.	5-58
:ANALysis:WPARameter<x>:TRENd: HRANge	Sets the target window for trend display of measured waveform parameters or queries the current setting.	5-58
:ANALysis:WPARameter<x>:TRENd: HSPan	Sets the horizontal span of the trend display of the waveform parameter measurement or queries the current setting.	5-58
:ANALysis:WPARameter<x>:TRENd: MEASure?	Queries all settings related to the automated measurement of the trend display of the waveform parameter measurement.	5-58

## 5.1 List of Commands

Command	Function	Page
:ANALysis:WPARAMeter<x>:TRENd:MEASure:CURSor?	Queries all settings related to the cursor measurement of the trend of the waveform parameter measurement.	5-58
:ANALysis:WPARAMeter<x>:TRENd:MEASure:CURSor:ALL	Turns ON/OFF all cursors of the trend of the waveform parameter measurement.	5-58
:ANALysis:WPARAMeter<x>:TRENd:MEASure:CURSor:C<x>?	Queries all settings related to the measured value of the cursor of the trend of the waveform parameter measurement.	5-58
:ANALysis:WPARAMeter<x>:TRENd:MEASure:CURSor:C<x>:POSITion	Sets the cursor position of the trend of the waveform parameter measurement or queries the current setting.	5-59
:ANALysis:WPARAMeter<x>:TRENd:MEASure:CURSor:C<x>:STATe	Turns ON/OFF the cursor of the trend of the waveform parameter measurement or queries the current setting.	5-59
:ANALysis:WPARAMeter<x>:TRENd:MEASure:CURSor:C<x>:VALue?	Queries the measured value of the cursor of the trend of the waveform parameter measurement.	5-59
:ANALysis:WPARAMeter<x>:TRENd:MEASure:CURSor:DC?	Queries all settings related to the measurement between cursors on the trend of the waveform parameter measurement.	5-59
:ANALysis:WPARAMeter<x>:TRENd:MEASure:CURSor:DC:STATe	Turns ON/OFF the measurement between cursors on the trend of the waveform parameter measurement or queries the current setting.	5-59
:ANALysis:WPARAMeter<x>:TRENd:MEASure:CURSor:DC:VALue?	Sets the measured value between cursors on the trend of the waveform parameter measurement or queries the current setting.	5-59
:ANALysis:WPARAMeter<x>:TRENd:MEASure:CURSor:LINKage	Turns ON/OFF the trend cursor link of the waveform parameter measurement or queries the current setting.	5-60
:ANALysis:WPARAMeter<x>:TRENd:MEASure:MODE	Sets the automated measurement mode of the trend of the waveform parameter measurement or queries the current setting.	5-60
:ANALysis:WPARAMeter<x>:TRENd:VERTical	Sets the vertical range of the trend of the waveform parameter measurement or queries the current setting.	5-60
:ANALysis:XY<x>?	Queries all settings related to the XY display function.	5-60
:ANALysis:XY<x>:GATE?	Queries all settings related to the gate function of the XY display.	5-60
:ANALysis:XY<x>:GATE:ALEVel	Sets the active level of the gate of the XY display or queries the current setting.	5-61
:ANALysis:XY<x>:GATE:HYSTeresis<x>	Sets the hysteresis of the gate of the XY display or queries the current setting.	5-61
:ANALysis:XY<x>:GATE:LEVel<x>	Sets the level of the gate of the XY display or queries the current setting.	5-61
:ANALysis:XY<x>:GATE:TRACe	Sets the gate trace of the XY display or queries the current setting.	5-61
:ANALysis:XY<x>:MEASure?	Queries all settings related to the automated measurement of the XY display.	5-61
:ANALysis:XY<x>:MEASure:CURSor?	Queries all settings related to the cursor measurement of the XY display.	5-61
:ANALysis:XY<x>:MEASure:CURSor:XLINKage	Turns ON/OFF the horizontal cursor link on the XY display or queries the current setting.	5-62
:ANALysis:XY<x>:MEASure:CURSor:X<x>?	Queries all settings related to the horizontal cursor of the XY display.	5-62
:ANALysis:XY<x>:MEASure:CURSor:X<x>:POSITion	Sets the horizontal cursor position of the XY display or queries the current setting.	5-62
:ANALysis:XY<x>:MEASure:CURSor:X<x>:VALue?	Queries the voltage value at the horizontal cursor of the XY display.	5-62
:ANALysis:XY<x>:MEASure:CURSor:YLINKage	Turns ON/OFF the vertical cursor link on the XY display or queries the current setting.	5-62
:ANALysis:XY<x>:MEASure:CURSor:Y<x>?	Queries all settings related to the vertical cursor of the XY display.	5-62
:ANALysis:XY<x>:MEASure:CURSor:Y<x>:POSITion	Sets the vertical cursor position of the XY display or queries the current setting.	5-62
:ANALysis:XY<x>:MEASure:CURSor:Y<x>:VALue?	Queries the voltage value at the vertical cursor of the XY display.	5-63
:ANALysis:XY<x>:MEASure:MODE	Sets the automated measurement mode of the XY display or queries the current setting.	5-63
:ANALysis:XY<x>:MEASure:XYINteg?	Queries all settings related to the integration of the XY display.	5-63
:ANALysis:XY<x>:MEASure:XYINteg:LOOP	Sets the integration mode of the XY display or queries the current setting.	5-63
:ANALysis:XY<x>:MEASure:XYINteg:POLarity	Sets the integration polarity of the XY display or queries the current setting.	5-63
:ANALysis:XY<x>:MEASure:XYINteg:VALue?	Queries the integral value of the XY display.	5-63
:ANALysis:XY<x>:TRANge (Time Range)	Sets the measurement range of the XY display or queries the current setting.	5-63
:ANALysis:XY<x>:WINDow	Sets the measurement source window of the XY display or queries the current setting.	5-63



Command	Function	Page
:ANALysis:XY<x>:XTRace	Sets the X-axis trace of the XY display or queries the current setting.	5-64
:ANALysis:XY<x>:YTRace	Sets the Y-axis trace of the XY display or queries the current setting.	5-64

**ASETup Group**

:ASETup:EXECute	Execute auto setup.	5-65
:ASETup:UNDO	Cancels auto setup that has been executed.	5-65

**CALibrate Group**

:CALibrate?	Queries all settings related to the calibration.	5-66
:CALibrate:EXECute	Executes calibration.	5-66
:CALibrate:MODE	Turns ON/OFF the auto calibration or queries the current setting.	5-66

**CHANnel Group**

:CHANnel<x>?	Queries all settings related to the channel.	5-67
:CHANnel<x>:ASCAle[:EXECute]	Executes the auto scaling of the channel.	5-67
:CHANnel<x>:BWIDth	Sets the input filter of the channel or queries the current setting.	5-67
:CHANnel<x>:COUPling	Sets the input coupling of the channel or queries the current setting.	5-67
:CHANnel<x>:DESKew	Sets the skew correction of the channel or queries the current setting.	5-67
:CHANnel<x>:DISPlay	Turns ON/OFF the display of the channel or queries the current setting.	5-67
:CHANnel<x>:INVert	Turns ON/OFF the inverted display of the channel or queries the current setting.	5-67
:CHANnel<x>:LABel?	Queries all settings related to the waveform label of the channel.	5-67
:CHANnel<x>:LABel[:DEFine]	Sets the waveform label of channel or queries the current setting.	5-67
:CHANnel<x>:LABel:MODE	Turns ON/OFF the waveform label display of the channel or queries the current setting.	5-67
:CHANnel<x>:OCANcel	Turns ON/OFF the offset cancel of the channel or queries the current setting.	5-68
:CHANnel<x>:OFFSet	Sets the offset voltage of the channel or queries the current setting.	5-68
:CHANnel<x>:POSItion	Sets the vertical position of the channel or queries the current setting.	5-68
:CHANnel<x>:PROBE?	Queries all settings related to the probe attenuation of the channel.	5-68
:CHANnel<x>:PROBE[:MODE]	Sets the probe attenuation of the channel or queries the current setting.	5-68
:CHANnel<x>:PROBE:AUTO?	Queries the probe attenuation of the channel when set to AUTO.	5-68
:CHANnel<x>:PZCalibrate[:EXECute]	Execute the zero adjustment of the current probe of the channel.	5-68
:CHANnel<x>:SELEct	Sets the waveform (input/computation) to be assigned to the input channel or queries the current setting.	5-68
:CHANnel<x>:SVALue (Scale VALUE)	Turns ON/OFF the scale display of the channel or queries the current setting.	5-68
:CHANnel<x>:VDIV	Sets the vertical sensitivity (V/div) of the channel or queries the current setting.	5-68

**CLEar Group**

:CLEar:ACCumulate	Clears accumulated waveforms.	5-69
:CLEar[:HISTory]	Clears history waveforms.	5-69
:CLEar:SNAP	Clears snapshot waveforms.	5-69

**COMMunicate Group**

:COMMunicate?	Queries all settings related to communications.	5-70
:COMMunicate:HEADer	Sets whether to add a header to the response to a query (example CHANNEL1:VOLTAGE:PROBE 10) or not add the header (example 10) or queries the current setting.	5-70
:COMMunicate:LOCKout	Sets or clears local lockout.	5-70
:COMMunicate:OPSE (Operation Pending Status Enable register)	Sets the overlap command that is used by the *OPC, *OPC?, and *WAI commands or queries the current setting.	5-70
:COMMunicate:OPSR? (Operation Pending Status Register)	Queries the value of the operation pending status register.	5-70
:COMMunicate:OVERlap	Sets the commands to operate as overlap commands or queries the current setting.	5-70
:COMMunicate:REMOte	Sets remote or local. ON is remote mode.	5-71
:COMMunicate:STATus?	Queries line-specific status.	5-71
:COMMunicate:VERBose	Sets whether to return the response to a query using full spelling (example CHANNEL1:VOLTAGE:PROBE 10) or using abbreviation (example CHAN:PROB 10) or queries the current setting.	5-71

## 5.1 List of Commands

Command	Function	Page
:COMMunicate:WAIT	Waits for one of the specified extended events to occur.	5-71
:COMMunicate:WAIT?	Creates the response that is returned when the specified event occurs.	5-71
<b>CURSOR Group</b>		
:CURSOR?	Queries all settings related to the cursor measurement.	5-72
:CURSOR:DISPlay	Turns ON/OFF the cursor or queries the current setting.	5-72
:CURSOR:HORizontal?	Queries all settings related to the horizontal cursors.	5-72
:CURSOR:HORizontal:BASic?	Queries all settings related to basic items of the horizontal cursors.	5-72
:CURSOR:HORizontal[:BASic]:ALL	Turns ON/OFF all basic items of the horizontal cursors.	5-72
:CURSOR:HORizontal[:BASic]:DV?	Queries all settings related to the $\Delta V$ measurement of the horizontal cursors.	5-72
:CURSOR:HORizontal[:BASic]:DV:STATE	Turns ON/OFF the $\Delta V$ measurement of the horizontal cursors or queries the current setting.	5-72
:CURSOR:HORizontal[:BASic]:DV:VALue?	Queries the voltage value between the horizontal cursors.	5-72
:CURSOR:HORizontal[:BASic]:LINKage	Turns ON/OFF the horizontal cursor link or queries the current setting.	5-72
:CURSOR:HORizontal[:BASic]:V<x>?	Queries all settings related to the horizontal cursor.	5-73
:CURSOR:HORizontal[:BASic]:V<x>:JUMP	Jumps the horizontal cursor to the center position of the zoom waveform.	5-73
:CURSOR:HORizontal[:BASic]:V<x>:POSITION	Sets the horizontal cursor position or queries the current setting.	5-73
:CURSOR:HORizontal[:BASic]:V<x>:STATE	Turns ON/OFF the horizontal cursor or queries the current setting.	5-73
:CURSOR:HORizontal[:BASic]:V<x>:VALue?	Queries the voltage value at the horizontal cursor.	5-73
:CURSOR:HORizontal:CALCulation?	Queries all settings related to calculation items of the horizontal cursors.	5-73
:CURSOR:HORizontal:CALCulation:ALL	Turns ON/OFF all calculation items of the horizontal cursors.	5-73
:CURSOR:HORizontal:CALCulation:DEFine<x>	Sets the equation of the calculation item of the horizontal cursor or queries the current setting.	5-73
:CURSOR:HORizontal:CALCulation:STATE<x>	Turns ON/OFF the calculation item of the horizontal cursor or queries the current setting.	5-73
:CURSOR:HORizontal:CALCulation:VALue<x>?	Queries the measured value of the calculation item of the horizontal cursor.	5-73
:CURSOR:HORizontal:TRACe	Sets the source trace of the horizontal cursor or queries the current setting.	5-74
:CURSOR:MARKer?	Queries all settings related to the marker cursors.	5-74
:CURSOR:MARKer:CALCulation?	Queries all settings related to calculation items of the marker cursors.	5-74
:CURSOR:MARKer:CALCulation:ALL	Turns ON/OFF all calculation items of the marker cursors.	5-74
:CURSOR:MARKer:CALCulation:DEFine<x>	Sets the equation of the calculation item of the marker cursors or queries the current setting.	5-74
:CURSOR:MARKer:CALCulation:STATE<x>	Turns ON/OFF the calculation item of the marker cursors or queries the current setting.	5-74
:CURSOR:MARKer:CALCulation:VALue<x>?	Queries the measured value of the calculation item of the marker cursors.	5-74
:CURSOR:MARKer:CM<x>?	Queries all settings related to the marker cursor.	5-74
:CURSOR:MARKer:CM<x>:ALL	Turns ON/OFF all items of the marker cursor.	5-75
:CURSOR:MARKer:CM<x>:DT<x>?	Queries all settings related to the $\Delta T$ measurement of the cursor marker.	5-75
:CURSOR:MARKer:CM<x>:DT<x>:STATE	Turns ON/OFF the $\Delta T$ measurement of the maker cursor or queries the current setting.	5-75
:CURSOR:MARKer:CM<x>:DT<x>:VALue?	Queries the $\Delta T$ value of the marker cursor.	5-75
:CURSOR:MARKer:CM<x>:DV<x>?	Queries all settings related to the $\Delta V$ measurement of the cursor marker.	5-75
:CURSOR:MARKer:CM<x>:DV<x>:STATE	Turns ON/OFF the $\Delta V$ measurement of the maker cursor or queries the current setting.	5-75
:CURSOR:MARKer:CM<x>:DV<x>:VALue?	Queries the $\Delta V$ value of the marker cursor.	5-75
:CURSOR:MARKer:CM<x>:JUMP	Jumps the marker cursor to the center position of the zoom waveform.	5-75
:CURSOR:MARKer:CM<x>:POSITION	Sets the marker cursor position or queries the current setting.	5-75
:CURSOR:MARKer:CM<x>:T?	Queries all settings related to the time measurement of the marker cursor.	5-75
:CURSOR:MARKer:CM<x>:T:STATE	Turns ON/OFF the time measurement of the maker cursor or queries the current setting.	5-76
:CURSOR:MARKer:CM<x>:T:VALue?	Queries the time value at the marker cursor position.	5-76

## 5.1 List of Commands

Command	Function	Page
:CURSor:MARKer:CM<x>:TRACe	Sets the source trace of the marker cursor or queries the current setting.	5-76
:CURSor:MARKer:CM<x>:V?	Queries all settings related to the voltage measurement of the marker cursor.	5-76
:CURSor:MARKer:CM<x>:V:STATe	Turns ON/OFF the voltage measurement of the maker cursor or queries the current setting.	5-76
:CURSor:MARKer:CM<x>:V:VALue?	Queries the voltage value at the marker cursor position.	5-76
:CURSor:MARKer:FORM	Sets the marker cursor form or queries the current setting.	5-76
:CURSor:SERial?	Queries all settings related to the serial cursors.	5-76
:CURSor:SERial:SCURsor<x>?	Queries all settings related to the serial cursor.	5-76
:CURSor:SERial:SCURsor<x>:ACTive	Sets the active level of the serial cursor or queries the current setting.	5-76
:CURSor:SERial:SCURsor<x>:BCOunt	Sets the bit length of the serial cursor or queries the current setting.	5-77
:CURSor:SERial:SCURsor<x>:BITRate	Sets the bit rate of the serial cursor or queries the current setting.	5-77
:CURSor:SERial:SCURsor<x>:BITOrder	Sets the bit order of the serial cursor or queries the current setting.	5-77
:CURSor:SERial:SCURsor<x>:FORMat	Sets the display format of the serial cursor or queries the current setting.	5-77
:CURSor:SERial:SCURsor<x>:HYSTEResis	Sets the hysteresis of the serial cursor or queries the current setting.	5-77
:CURSor:SERial:SCURsor<x>:JUMP	Moves the serial cursor to the specified direction.	5-77
:CURSor:SERial:SCURsor<x>:LEVel	Sets the threshold level of the serial cursor or queries the current setting.	5-77
:CURSor:SERial:SCURsor<x>:MODE	Turns ON/OFF the serial cursor or queries the current setting.	5-77
:CURSor:SERial:SCURsor<x>:POSITION	Sets the serial cursor position or queries the current setting.	5-77
:CURSor:SERial:SCURsor<x>:TRACe	Sets the trace of the serial cursor or queries the current setting.	5-78
:CURSor:SERial:SCURsor<x>:TRACk	Jumps the serial cursor onto the zoom waveform.	5-78
:CURSor:SERial:SCURsor<x>:VALue?	Queries the measured value of the serial cursor.	5-78
:CURSor:TYPE	Sets the cursor type or queries the current setting.	5-78
:CURSor:VERTical?	Queries all settings related to the vertical cursors.	5-78
:CURSor:VERTical:BASic?	Queries all settings related to basic items of the vertical cursors.	5-78
:CURSor:VERTical[:BASic]:ALL	Turns ON/OFF all basic items of the vertical cursors.	5-78
:CURSor:VERTical[:BASic]:DT?	Queries all settings related to the $\Delta T$ measurement of the vertical cursors.	5-78
:CURSor:VERTical[:BASic]:DT:STATe	Turns ON/OFF the $\Delta T$ measurement of the vertical cursors or queries the current setting.	5-78
:CURSor:VERTical[:BASic]:DT:VALue?	Queries the $\Delta T$ value of the vertical cursors.	5-78
:CURSor:VERTical[:BASic]:LINKage	Turns ON/OFF the vertical cursor link or queries the current setting.	5-78
:CURSor:VERTical[:BASic]:PERDt?	Queries all settings related to the $1/\Delta T$ measurement of the vertical cursors.	5-79
:CURSor:VERTical[:BASic]:PERDt:STATe	Turns ON/OFF the $1/\Delta T$ measurement of the vertical cursors or queries the current setting.	5-79
:CURSor:VERTical[:BASic]:PERDt:VALue?	Queries the $1/\Delta T$ value of the vertical cursors.	5-79
:CURSor:VERTical[:BASic]:T<x>?	Queries all settings related to the vertical cursor.	5-79
:CURSor:VERTical[:BASic]:T<x>:JUMP	Jumps to the center position of the zoom waveform of the vertical cursor.	5-79
:CURSor:VERTical[:BASic]:T<x>:POSITION	Sets the vertical cursor position or queries the current setting.	5-79
:CURSor:VERTical[:BASic]:T<x>:STATe	Turns ON/OFF the vertical cursor or queries the current setting.	5-79
:CURSor:VERTical[:BASic]:T<x>:VALue?	Queries the time value at the vertical cursor.	5-79
:CURSor:VERTical:CALCulation?	Queries all settings related to calculation items of the vertical cursors.	5-79
:CURSor:VERTical:CALCulation:ALL	Turns ON/OFF all calculation items of the vertical cursors.	5-79
:CURSor:VERTical:CALCulation:DEFine<x>	Sets the equation of the calculation item of the vertical cursor or queries the current setting.	5-80
:CURSor:VERTical:CALCulation:STATe<x>	Turns ON/OFF the calculation item of the vertical cursor or queries the current setting.	5-80
:CURSor:VERTical:CALCulation:VALue<x>?	Queries the measured value of the calculation item of the vertical cursor.	5-80

## 5.1 List of Commands

Command	Function	Page
:CURSor:VT?	Queries all settings related to the VT cursor.	5-80
:CURSor:VT:BASic?	Queries all settings related to basic items of the VT cursor.	5-80
:CURSor:VT[:BASic]:ALL	Turns ON/OFF all basic items of the VT cursor.	5-80
:CURSor:VT[:BASic]:POSition	Sets the VT cursor position or queries the current setting.	5-80
:CURSor:VT[:BASic]:T?	Queries all settings related to the time value of the VT cursor.	5-80
:CURSor:VT[:BASic]:T:STATe	Turns ON/OFF the time value of the VT cursor or queries the current setting.	5-80
:CURSor:VT[:BASic]:T:VALue?	Queries the time value at the VT cursor.	5-80
:CURSor:VT[:BASic]:V<x>?	Queries all settings related to the voltage value of the VT cursor.	5-81
:CURSor:VT[:BASic]:V<x>:STATe	Turns ON/OFF the voltage value of the VT cursor or queries the current setting.	5-81
:CURSor:VT[:BASic]:V<x>:VALue?	Queries the voltage value at the VT cursor.	5-81
:CURSor:VT:CALCulation?	Queries all settings related to calculation items of the VT cursor.	5-81
:CURSor:VT:CALCulation:ALL	Turns ON/OFF all calculation items of the VT cursor.	5-81
:CURSor:VT:CALCulation:DEFine<x>	Sets the equation of the calculation item of the VT cursor or queries the current setting.	5-81
:CURSor:VT:CALCulation:STATe<x>	Turns ON/OFF the calculation item of the VT cursor or queries the current setting.	5-81
:CURSor:VT:CALCulation:VALue<x>?	Queries the measured value of the calculation item of the VT cursor.	5-81
:CURSor:VT:JUMP	Jumps the VT cursor to the center position of the zoom waveform.	5-81

### DISPlay Group

:DISPlay?	Queries all settings related to the display.	5-82
:DISPlay:ACCumulate?	Queries all settings related to the accumulated display of waveforms.	5-82
:DISPlay:ACCumulate:GRADE	Sets the accumulate mode or queries the current setting.	5-82
:DISPlay:ACCumulate:MODE	Turns ON/OFF the accumulate mode or queries the current setting.	5-82
:DISPlay:ACCumulate:PERSistence?	Queries all settings related to persistence.	5-82
:DISPlay:ACCumulate:PERSistence:COUNT	Sets the persistence count or queries the current setting.	5-82
:DISPlay:ACCumulate:PERSistence:MODE	Sets the persistence mode or queries the current setting.	5-82
:DISPlay:ACCumulate:PERSistence:TIME	Sets the persistence time or queries the current setting.	5-82
:DISPlay:BLIGHt?	Queries all settings related to the backlight.	5-83
:DISPlay:BLIGHt:AUTOoff	Sets the function that automatically turns the backlight off or queries the current setting.	5-83
:DISPlay:BLIGHt:BRIGHtness	Sets the brightness of the backlight or queries the current setting.	5-83
:DISPlay:BLIGHt:LCD	Turns ON/OFF the backlight or queries the current setting.	5-83
:DISPlay:BLIGHt:TIMEout	Sets the timeout of the backlight or queries the current setting.	5-83
:DISPlay:COLor?	Queries all settings related to the waveform display color.	5-83
:DISPlay:COLor:{CHANnel<x> MATH<x> REFerence<x>}	Sets the color of the waveform or queries the current setting.	5-83
:DISPlay:FORMat	Sets the display format or queries the current setting.	5-83
:DISPlay:GRATICule	Sets the graticule (grid) or queries the current setting.	5-83
:DISPlay:INTENSity?	Queries all settings related to the intensity of the displayed items.	5-83
:DISPlay:INTENSity[:WAVEform]	Sets the intensity of the waveform or queries the current setting.	5-83
:DISPlay:INTENSity:{CURSor GRID MARKer ZBOX}	Sets the intensity of the display item or queries the current setting.	5-84
:DISPlay:INTERpolate	Sets the display interpolation format or queries the current setting.	5-84
:DISPlay:MAPPING?	Queries all settings related to the waveform mapping to the split screen.	5-84
:DISPlay:MAPPING[:MODE]	Sets the waveform mapping mode for the split screen or queries the current setting.	5-84
:DISPlay:MAPPING:TRACe<x>	Sets the mapping of the waveform to the split screen or queries the current setting.	5-84
:DISPlay:TRANslucent	Turns ON/OFF the translucent mode or queries the current setting.	5-84

### FILE Group

:FILE?	Queries all settings related to the specified storage medium or internal memory.	5-85
:FILE:DIRectory?	Queries all settings related to the specified storage medium.	5-85

Command	Function	Page
:FILE[:DIRectory]:CDIRectory (Change Directory)	Sets the current directory or queries the current setting.	5-85
:FILE[:DIRectory]:FREE?	Queries the free disk space in bytes on the specified storage medium.	5-85
:FILE[:DIRectory]:MDIRectory (Make Directory)	Creates a directory under the specified directory. This is an overlap command.	5-86
:FILE:INTernal?	Queries all settings related to the internal memory.	5-86
:FILE:INTernal:RECall:DMEMory[: EXECute]	Executes the loading of the accumulated data from the internal memory. This is an overlap command.	5-86
:FILE:INTernal:RECall:SETup[: EXECute]	Executes the loading of the setup data from the internal memory. This is an overlap command.	5-86
:FILE:INTernal:STORe?	Queries all settings related to the saving to the internal memory.	5-86
:FILE:INTernal:STORe:BINary?	Queries all settings related to the saving of the binary data to the internal memory.	5-86
:FILE:INTernal:STORe:BINary[: EXECute]	Executes the saving of the binary data to the internal memory. This is an overlap command.	5-86
:FILE:INTernal:STORe:BINary: TRACe	Sets the trace to be saved in binary data to the internal memory or queries the current setting.	5-86
:FILE:INTernal:STORe:DMEMory?	Queries all settings related to the saving of the accumulated data to the internal memory.	5-86
:FILE:INTernal:STORe:DMEMory[: EXECute]	Executes the saving of the accumulated data to the internal memory. This is an overlap command.	5-87
:FILE:INTernal:STORe:DMEMory: TRACe	Sets the trace to be saved in accumulated data to the internal memory or queries the current setting.	5-87
:FILE:INTernal:STORe:SETup?	Queries all settings related to the saving of the setup data to the internal memory.	5-87
:FILE:INTernal:STORe:SETup: COMMeNt<x>	Sets the comment to the setup data to be saved to the internal memory or queries the current setting.	5-87
:FILE:INTernal:STORe:SETup: DATE<x>?	Queries the date/time when the setup data was saved to the internal memory.	5-87
:FILE:INTernal:STORe:SETup[: EXECute]	Executes the saving of the setup data to the internal memory. This is an overlap command.	5-87
:FILE:INTernal:STORe:SETup: LOCK<x>	Turns ON/OFF the read-only attribute of the setup data in the internal memory or queries the current setting.	5-87
:FILE:INTernal:UNLoad:DMEMory[: EXECute]	Clears the loaded accumulated data.	5-87
:FILE:LOAD?	Queries all settings related to the loading of files on the specified storage medium.	5-87
:FILE:LOAD:{BINary ZPOLYgon ZWA Ve}?	Queries all settings related to the loading of specific data.	5-87
:FILE:LOAD:{BINary DMEMory MASK  SETup ZPOLYgon ZWAVE}:ABORt	Aborts the loading of specific data.	5-87
:FILE:LOAD:{BINary DMEMory  MASK SETup ZPOLYgon ZWAVE}[: EXECute]	Executes the loading of specific data. This is an overlap command.	5-88
:FILE:LOAD:BINary:REFerence	Sets the load destination of the binary data or queries the current setting.	5-88
:FILE:LOAD:{ZPOLYgon ZWAVE}: ZONE	Sets the load destination of the zone data of specific data or queries the current setting.	5-88
:FILE:SAVE?	Queries all settings related to the saving of data.	5-88
:FILE:SAVE:{AHISTogram ASCIi BI Nary DMEMory FFT FLOat SBUS ZWA Ve}?	Queries all settings related to the saving of specific data.	5-88
:FILE:SAVE:{AHISTogram ASCIi BI Nary DMEMory FFT FLOat SETup SB US WPARAMeter ZWAVE}:ABORt	Aborts the save operation of specific data.	5-88
:FILE:SAVE:{AHISTogram ASCIi BI Nary DMEMory FFT FLOat SETup SB US WPARAMeter ZWAVE}[:EXECute]	Executes the saving of specific data by specifying a file name. This is an overlap command.	5-88
:FILE:SAVE:{AHISTogram FFT SBUS }:ANALYsis	Sets the trace of specific data to be saved or queries the current setting.	5-89
:FILE:SAVE:ANAMing	Sets the type of auto naming of file names of the saved data or queries the current setting.	5-89
:FILE:SAVE:ASCIi:RANGe	Sets the ASCII data saving range or queries the current setting.	5-89
:FILE:SAVE:{ASCIi BINary FLOat} :COMPRession	Sets the compression method for saving specific data or queries the current setting.	5-89

## 5.1 List of Commands

Command	Function	Page
:FILE:SAVE:{ASCIi BINary FLOat} :HISTory	Sets whether to save the entire data or the selected data of specific data or queries the current setting.	5-89
:FILE:SAVE:{ASCIi BINary FLOat} :LENGth	Sets the size of waveform data to save for each type of data or queries the current setting.	5-89
:FILE:SAVE:{ASCIi BINary FLOat} :TRACe	Sets the trace of specific data to be saved or queries the current setting.	5-89
:FILE:SAVE:COMMeNt	Sets the comment of data to be saved or queries the current setting.	5-89
:FILE:SAVE:DMEMory:TRACe	Sets the trace to be saved as accumulated data or queries the current setting.	5-90
:FILE:SAVE:NAME	Sets the name of the data file to be saved or queries the current setting.	5-90
:FILE:SAVE:ZWAVE:ZONE	Sets the zone of the zone data to be saved or queries the current setting.	5-90

### GONogo Group

:GONogo?	Queries all settings related to GO/NO-GO determination.	5-91
:GONogo:ABORt	Aborts the GO/NO-GO determination.	5-91
:GONogo:ACTion?	Queries all settings related to the action taken when the determination result is NO-GO and the criteria values.	5-91
:GONogo:ACTion:BUZZer	Sets whether to sound a buzzer when the determination result is NO-GO or queries the current setting.	5-91
:GONogo:ACTion:HCOPy	Sets whether to print the screen image on the printer when the determination result is NO-GO or queries the current setting.	5-91
:GONogo:ACTion:MAIL?	Queries all settings related to the mail transmission when the determination is NO-GO.	5-91
:GONogo:ACTion:MAIL:INTERval	Sets the interval at which to send mail when the determination is NO-GO or queries the current setting.	5-91
:GONogo:ACTion:MAIL:MODE	Sets whether to send mail when the determination is NO-GO or queries the current setting.	5-91
:GONogo:ACTion:SAVE	Sets whether to save the waveform data to the storage medium when the determination result is NO-GO or queries the current setting.	5-92
:GONogo:CONDition<x>	Sets the GO/NO-GO determination criteria or queries the current setting.	5-92
:GONogo:COUNT?	Queries the actual number of GO/NO-GO determinations.	5-92
:GONogo:EXECute	Executes the GO/NO-GO determination. This is an overlap command.	5-92
:GONogo:LOGic	Sets the GO/NO-GO determination logic or queries the current setting.	5-92
:GONogo:MODE	Sets the GO/NO-GO determination type or queries the current setting.	5-92
:GONogo:NGCount?	Queries the actual number of NO-GOs of the GO/NO-GO determination.	5-92
:GONogo:SCONdition? (Stop Condition)	Queries all settings related to the determination termination condition.	5-92
:GONogo[:SCONdition]:NGCount	Sets the number of NO-GOs that terminates the GO/NO-GO determination or queries the current setting.	5-92
:GONogo[:SCONdition]:STOPcount	Sets the acquisition count that terminates the GO/NO-GO determination or queries the current setting.	5-92
:GONogo:TELecomtest?	Queries all settings related to telecom test determination.	5-92
:GONogo:TELecomtest:SElect<x>?	Queries all settings related to the condition of the telecom test determination.	5-92
:GONogo:TELecomtest:SElect<x>: EYEPattern?	Queries all settings related to the eye pattern determination of the condition.	5-93
:GONogo:TELecomtest:SElect<x>: EYEPattern:<Parameter>	Sets the upper and lower limits of the waveform parameter of the eye pattern determination or queries the current setting.	5-93
:GONogo:TELecomtest:SElect<x>: MASK?	Queries all settings related to the mask determination of the condition.	5-93
:GONogo:TELecomtest:SElect<x>: MASK:ELEMent<x>?	Queries all settings related to the element used in the mask determination.	5-93
:GONogo:TELecomtest:SElect<x>: MASK:ELEMent<x>:PSPCount (Sample Point Count %)	Sets the upper and lower limits of the error rate for the number of sampled data points of the element or queries the current setting.	5-93
:GONogo:TELecomtest:SElect<x>: MASK:ELEMent<x>:PWCount (Wave Count %)	Sets the upper and lower limits of the error rate for the number of acquisitions of the element or queries the current setting.	5-93
:GONogo:TELecomtest:SElect<x>: MASK:ELEMent<x>:SPCount (Sample Point Count)	Sets the upper and lower limits of the number of sampled data points for the element that results in error or queries the current setting.	5-94
:GONogo:TELecomtest:SElect<x>: MASK:ELEMent<x>:WCount (Wave Count)	Sets the upper and lower limits of the number of acquisitions for the element that results in error or queries the current setting.	5-94

## 5.1 List of Commands

Command	Function	Page
:GONogo:ZPARAMeter?	Queries all settings related to zone/parameter determination.	5-94
:GONogo:ZPARAMeter:SElect<x>?	Queries all settings related to the condition of the zone/parameter determination.	5-94
:GONogo:ZPARAMeter:SElect<x>: MODE	Sets the mode of the condition or queries the current setting.	5-94
:GONogo:ZPARAMeter:SElect<x>: PARAMeter?	Queries all settings related to the condition parameter.	5-95
:GONogo:ZPARAMeter:SElect<x>: PARAMeter:CATegory	Sets the parameter category or queries the current setting.	5-95
:GONogo:ZPARAMeter:SElect<x>: PARAMeter:FFT<x>?	Queries all settings related to the FFT determination.	5-95
:GONogo:ZPARAMeter:SElect<x>: PARAMeter:FFT<x>:CALCulation<x>	Sets the upper and lower limits of the calculation item of the FFT determination or queries the current setting.	5-95
:GONogo:ZPARAMeter:SElect<x>: PARAMeter:FFT<x>:PEAK?	Queries all settings related to the peak value of the FFT determination.	5-95
:GONogo:ZPARAMeter:SElect<x>: PARAMeter:FFT<x>:PEAK: DFREquency	Sets the upper and lower limits between the peak frequencies of the FFT determination or queries the current setting.	5-95
:GONogo:ZPARAMeter:SElect<x>: PARAMeter:FFT<x>:PEAK:DV	Sets the upper and lower limits between the peak voltages of the FFT determination or queries the current setting.	5-96
:GONogo:ZPARAMeter:SElect<x>: PARAMeter:FFT<x>:PEAK: FREquency<x>	Sets the upper and lower limits of the peak frequency of the FFT determination or queries the current setting.	5-96
:GONogo:ZPARAMeter:SElect<x>: PARAMeter:FFT<x>:PEAK:V<x>	Sets the upper and lower limits of the peak voltage of the FFT determination or queries the current setting.	5-96
:GONogo:ZPARAMeter:SElect<x>: PARAMeter:MEASure?	Queries all settings related to the determination using automated measurement of waveform parameters (measure determination).	5-96
:GONogo:ZPARAMeter: SElect<x>:PARAMeter:MEASure: CALCulation<x>	Sets the upper and lower limits of the calculation item of the measure determination or queries the current setting.	5-96
:GONogo:ZPARAMeter:SElect<x>: PARAMeter:MEASure:STATistics	Sets the statistical value of the measure determination or queries the current setting.	5-97
:GONogo:ZPARAMeter:SElect<x>: PARAMeter:MEASure:TRACe<x>?	Queries all settings related to the trace of the measure determination.	5-97
:GONogo:ZPARAMeter:SElect<x>: PARAMeter:MEASure:TRACe<x>: AREA<x>?	Queries all settings related to the area of the measure determination.	5-97
:GONogo:ZPARAMeter:SElect<x>: PARAMeter:MEASure:TRACe<x>: AREA<x>:TYPE?	Queries all settings related to the waveform parameters of the measure determination.	5-97
:GONogo:ZPARAMeter:SElect<x>: PARAMeter:MEASure:TRACe<x>: AREA<x>:TYPE:<Parameter>	Sets the upper and lower limits of the waveform parameter of the measure determination or queries the current setting.	5-97
:GONogo:ZPARAMeter:SElect<x>: PARAMeter:XY<x>?	Queries all settings related to the XY determination.	5-98
:GONogo:ZPARAMeter:SElect<x>: PARAMeter:XY<x>:XYINteg	Sets the upper and lower limits integral value of the XY determination or queries the current setting.	5-98
:GONogo:ZPARAMeter:SElect<x>: RECTangle?	Queries all settings related to the rectangle determination.	5-98
:GONogo:ZPARAMeter:SElect<x>: RECTangle:HORizontal	Sets the horizontal position of the rectangle used in the rectangle determination or queries the current setting.	5-98
:GONogo:ZPARAMeter:SElect<x>: RECTangle:VERTical	Sets the vertical position of the rectangle used in the rectangle determination or queries the current setting.	5-98
:GONogo:ZPARAMeter:SElect<x>: TRACe	Sets the source trace of the zone/parameter determination or queries the current setting.	5-98
:GONogo:ZPARAMeter:SElect<x>: WAVE?	Queries all settings related to the wave determination.	5-98
:GONogo:ZPARAMeter:SElect<x>: WAVE:EDIT<x>:EXIT	Exits the edit menu of the wave determination zone.	5-99
:GONogo:ZPARAMeter:SElect<x>: WAVE:EDIT<x>:PART	Editing the portion of the zone of the wave determination.	5-99
:GONogo:ZPARAMeter:SElect<x>: WAVE:EDIT<x>:WHOLe	Sets the editing of the entire zone of the waveform zone.	5-99
:GONogo:ZPARAMeter:SElect<x>: WAVE:TRANge	Sets the determination range of the zone determination or queries the current setting.	5-99

## 5.1 List of Commands

Command	Function	Page
:GONogo:ZPARAMeter:SElect<x>: WINDow	Sets the source window of the zone determination or queries the current setting.	5-99
<b>HCOPY Group</b>		
:HCOPY? (Hard COPY)	Queries all settings related to the output of screen data.	5-100
:HCOPY:ABORT	Aborts data output and paper feeding.	5-100
:HCOPY:DIRection	Sets the data output destination or queries the current setting.	5-100
:HCOPY:EXECute	Executes the data output. This is an overlap command.	5-100
:HCOPY:EXTPrinter?	Queries all settings related to the external printer output.	5-100
:HCOPY:EXTPrinter:TONE	Sets the half tone of the external printer output or queries the current setting.	5-100
:HCOPY:EXTPrinter:TYPE	Sets the type of output commands to send to the external printer or queries the current setting.	5-100
:HCOPY:FILE?	Queries all settings related to file output.	5-100
:HCOPY:FILE:FORMat	Sets the file output image format or queries the current setting.	5-100
:HCOPY:FILE:SAVE?	Queries all settings related to the saving of file output.	5-100
:HCOPY:FILE:SAVE:ANAMing	Sets the type of auto naming of save destination file names or queries the current setting.	5-101
:HCOPY:FILE:SAVE:CDIRectory (Change Directory)	Sets the save destination directory name or queries the current setting.	5-101
:HCOPY:FILE:SAVE:NAME	Sets the save destination file name or queries the current setting.	5-101
:HCOPY:NETPrint?	Queries all settings related to network printer output or queries the current setting.	5-101
:HCOPY:NETPrint:TONE	Sets the half tone for the network printer or queries the current setting.	5-101
:HCOPY:NETPrint:TYPE	Sets the output command type for the network printer or queries the current setting.	5-101
:HCOPY:PRINter?	Queries all settings related to the built-in printer output.	5-101
:HCOPY:PRINter:HRMode	Turns ON/OFF the high resolution mode of the built-in printer output or queries the current setting.	5-101
<b>HISTory Group</b>		
:HISTory?	Queries all settings related to the history function.	5-102
:HISTory:CURRent?	Queries all settings related to the history function of the current waveform (CH1 to 4, M1 to 8).	5-102
:HISTory[:CURRent]:DISPlay	Sets the start number and end number of the display record of the history waveform or queries the current setting.	5-102
:HISTory[:CURRent]:DMODE (Display Mode)	Sets the display mode of the history waveform or queries the current setting.	5-103
:HISTory[:CURRent]:MODE	Sets the highlight display mode of the history waveform or queries the current setting.	5-103
:HISTory[:CURRent]:RECORD	Sets the target record of the history waveform or queries the current setting.	5-103
:HISTory[:CURRent]:RECORD? MINimum	Queries the minimum record number of the history waveform.	5-103
:HISTory[:CURRent]:REPLay?	Queries all settings related to the replay function of the history function.	5-103
:HISTory[:CURRent]:REPLay:JUMP	Jumps the history waveform to the specified record number.	5-103
:HISTory[:CURRent]:REPLay:SPEEd	Sets the replay speed of the history waveform or queries the current setting.	5-103
:HISTory[:CURRent]:REPLay:STARt	Starts the replay of the history waveform in the specified direction.	5-103
:HISTory[:CURRent]:REPLay:STOP	Stops the replay of the history waveform.	5-103
:HISTory[:CURRent]:SEARCh?	Queries all settings related to the history search function.	5-104
:HISTory[:CURRent][:SEARCh]: ABORT	Aborts the history search.	5-104
:HISTory[:CURRent][:SEARCh]: EXECute	Executes the history search. This is an overlap command.	5-104
:HISTory[:CURRent][:SEARCh]: LOGic	Sets the history search logic or queries the current setting.	5-104
:HISTory[:CURRent][:SEARCh]: RESEt	Resets the search conditions of the history search.	5-104
:HISTory[:CURRent][:SEARCh]: SElect<x>?	Queries all settings related to the history search condition.	5-104
:HISTory[:CURRent][:SEARCh]: SElect<x>:CONDition	Sets determination criteria of the history search condition or queries the current setting.	5-104
:HISTory[:CURRent][:SEARCh]: SElect<x>:MODE	Sets the mode of the history search condition or queries the current setting.	5-104



## 5.1 List of Commands

Command	Function	Page
:HISTory[:CURRent] [:SEARch] : SElect<x>:PARAmeter?	Queries all settings related to the parameter of the history search condition.	5-105
:HISTory[:CURRent] [:SEARch] : SElect<x>:PARAmeter:CATegory	Sets the parameter category or queries the current setting.	5-105
:HISTory[:CURRent] [:SEARch] : SElect<x>:PARAmeter:FFT<x>?	Queries all settings related to the FFT search.	5-105
:HISTory[:CURRent] [:SEARch] : SElect<x>:PARAmeter:FFT<x>: CALCulation<x>	Sets the upper and lower limits of the calculation item of the FFT search or queries the current setting.	5-105
:HISTory[:CURRent] [:SEARch] : SElect<x>:PARAmeter:FFT<x>: PEAK?	Queries all settings related to the peak value of the FFT search.	5-105
:HISTory[:CURRent] [:SEARch] : SElect<x>:PARAmeter:FFT<x>: PEAK:DFREquency	Sets the upper and lower limits between the peak frequencies of the FFT search or queries the current setting.	5-106
:HISTory[:CURRent] [:SEARch] : SElect<x>:PARAmeter:FFT<x>: PEAK:DV	Sets the upper and lower limits between the peak voltages of the FFT search or queries the current setting.	5-106
:HISTory[:CURRent] [:SEARch] : SElect<x>:PARAmeter:FFT<x>: PEAK:FREquency<x>	Sets the upper and lower limits of the peak frequency of the FFT search or queries the current setting.	5-106
:HISTory[:CURRent] [:SEARch] : SElect<x>:PARAmeter:FFT<x>: PEAK:V<x>	Sets the upper and lower limits of the peak voltage of the FFT search or queries the current setting.	5-106
:HISTory[:CURRent] [:SEARch] : SElect<x>:PARAmeter:MEASure?	Queries all settings related to the search using automated measurement of waveform parameters (measure search).	5-106
:HISTory[:CURRent] [:SEARch] : SElect<x>:PARAmeter:MEASure: CALCulation<x>	Sets the upper and lower limits of the calculation item of the measure search or queries the current setting.	5-107
:HISTory[:CURRent] [:SEARch] : SElect<x>:PARAmeter:MEASure: TRACe<x>?	Queries all settings related to the trace of the measure search.	5-107
:HISTory[:CURRent] [:SEARch] : SElect<x>:PARAmeter:MEASure: TRACe<x>:AREA<x>?	Queries all settings related to the area of the measure search.	5-107
:HISTory[:CURRent] [:SEARch] : SElect<x>:PARAmeter:MEASure: TRACe<x>:AREA<x>:TYPE?	Queries all settings related to the waveform parameters of the measure search.	5-107
:HISTory[:CURRent] [:SEARch] : SElect<x>:PARAmeter: MEASure:TRACe<x>:AREA<x>: TYPE:<Parameter>	Sets the upper and lower limits of the waveform parameter of the measure search or queries the current setting.	5-108
:HISTory[:CURRent] [:SEARch] : SElect<x>:PARAmeter:XY<x>?	Queries all settings related to the XY search.	5-108
:HISTory[:CURRent] [:SEARch] : SElect<x>:PARAmeter:XY<x>: XYINteg	Sets the upper and lower limits integral value of the XY search or queries the current setting.	5-108
:HISTory[:CURRent] [:SEARch] : SElect<x>:RECTangle?	Queries all settings related to the rectangle search.	5-108
:HISTory[:CURRent] [:SEARch] : SElect<x>:RECTangle:HORizontal	Sets the horizontal position of the rectangle used in the rectangle search or queries the current setting.	5-108
:HISTory[:CURRent] [:SEARch] : SElect<x>:RECTangle:VERTical	Sets the vertical position of the rectangle used in the rectangle search or queries the current setting.	5-109
:HISTory[:CURRent] [:SEARch] : SElect<x>:TRACe	Sets the source trace of the history search or queries the current setting.	5-109
:HISTory[:CURRent] [:SEARch] : SElect<x>:WAVE?	Queries all settings related to the search in the waveform zone (wave search).	5-109
:HISTory[:CURRent] [:SEARch] : SElect<x>:WAVE:EDIT<x>:EXIT	Exits the zone edit menu of the wave search.	5-109
:HISTory[:CURRent] [:SEARch] : SElect<x>:WAVE:EDIT<x>:PART	Sets the editing of the portion of the zone of the wave search.	5-109
:HISTory[:CURRent] [:SEARch] : SElect<x>:WAVE:EDIT<x>:WHOLe	Sets the editing of the entire zone of the wave search.	5-109
:HISTory[:CURRent] [:SEARch] : SElect<x>:WAVE:TRANge	Sets the range over which to perform the wave search or queries the current setting.	5-110

## 5.1 List of Commands

Command	Function	Page
:HISTory[:CURRent] [:SEARch]:SELEct<x>:WINDow	Sets the source window of the history search or queries the current setting.	5-110
:HISTory[:CURRent]:TIME?	Queries the time of the source record number of the history waveform.	5-110
:HISTory:REFEreNce<x>?	Queries all settings related to the history function of the reference.	5-110
:HISTory:REFEreNce<x>:DMODE (Display Mode)	Sets the display mode of the history waveform of the reference or queries the current setting.	5-110
:HISTory:REFEreNce<x>:MODE	Sets the highlight display mode of the history waveform of the reference or queries the current setting.	5-110
:HISTory:REFEreNce<x>:RECOrd	Sets the source record of the history waveform of the reference or queries the current setting.	5-110
:HISTory:REFEreNce<x>:RECOrd? MINimum	Queries the minimum record number of the history waveform of the reference.	5-110
:HISTory:REFEreNce<x>:REPLAy?	Queries all settings related to the replay function of the history function of the reference.	5-110
:HISTory:REFEreNce<x>:REPLAy: JUMP	Jumps to the specified record number of the history waveform of the reference.	5-111
:HISTory:REFEreNce<x>:REPLAy: SPEEd	Sets the replay speed of the history waveform of the reference or queries the current setting.	5-111
:HISTory:REFEreNce<x>:REPLAy: START	Starts the replay of the history waveform of the reference.	5-111
:HISTory:REFEreNce<x>:REPLAy: STOP	Stops the replay of the history waveform of the reference.	5-111
:HISTory:REFEreNce<x>:TIME?	Queries the time of the source record number of the reference waveform.	5-111
<b>IMAGe Group</b>		
:IMAGe?	Queries all settings related to the output of screen image data.	5-112
:IMAGe:FORMAt	Sets the output format of the screen image data or queries the current setting.	5-112
:IMAGe:SEND?	Queries the screen image data.	5-112
:IMAGe:TONE	Sets the tone of the screen image data or queries the current setting.	5-112
<b>INITialize Group</b>		
:INITialize:EXECute	Execute the initialization.	5-113
:INITialize:UNDO	Cancels the initialization that was executed.	5-113
<b>MATH Group</b>		
:MATH<x>?	Queries all settings related to the computation.	5-114
:MATH<x>:DISPLAy	Turns ON/OFF the computed waveform or queries the current setting.	5-114
:MATH<x>:ECOUNt?	Queries all settings related to the edge count computation.	5-114
:MATH<x>:ECOUNt:HYSterEsis	Sets the hysteresis of the edge detection level of the edge count computation or queries the current setting.	5-114
:MATH<x>:ECOUNt:POLarity	Sets the edge detection polarity of the edge count computation or queries the current setting.	5-114
:MATH<x>:FILTer?	Queries all settings related to the filter.	5-114
:MATH<x>:FILTer:DELay?	Queries all settings related to the delay computation.	5-114
:MATH<x>:FILTer:DELay:TIME	Sets the delay value of the delay computation or queries the current setting.	5-114
:MATH<x>:FILTer:IIR?	Queries all settings related to the IIR filter computation.	5-115
:MATH<x>:FILTer:IIR:FORDer (Filter Order)	Sets the filter order of the IIR filter computation or queries the current setting.	5-115
:MATH<x>:FILTer:IIR:HIPass?	Queries all settings related to the IIR high pass filter computation.	5-115
:MATH<x>:FILTer:IIR:HIPass:COFF	Sets the cutoff frequency of the IIR high pass filter computation or queries the current setting.	5-115
:MATH<x>:FILTer:IIR:LOWPass?	Queries all settings related to the IIR low pass filter computation.	5-115
:MATH<x>:FILTer:IIR:LOWPass:COFF	Sets the cutoff frequency of the IIR low pass filter computation or queries the current setting.	5-115
:MATH<x>:FILTer:MAVG?	Queries all settings related to the moving average computation.	5-115
:MATH<x>:FILTer:MAVG:WEIGHt	Sets the weight of the moving average computation or queries the current setting.	5-115
:MATH<x>:FILTer:RESCAling?	Queries all settings related to the rescaling of the filter.	5-115
:MATH<x>:FILTer:RESCAling:AVALue	Sets rescaling coefficient A of the filter or queries the current setting.	5-115

## 5.1 List of Commands

Command	Function	Page
:MATH<x>:FILTer:RESCaling: BVALue	Sets rescaling offset B of the filter or queries the current setting.	5-116
:MATH<x>:FILTer:TYPE	Sets the filter type or queries the current setting.	5-116
:MATH<x>:INTEgral?	Queries all settings related to the integral computation.	5-116
:MATH<x>:INTEgral:PSCaling?	Queries all settings related to the pre-scaling of the integral computation.	5-116
:MATH<x>:INTEgral:PSCaling: AVALue	Sets pre-scaling coefficient A of the integral computation or queries the current setting.	5-116
:MATH<x>:INTEgral:PSCaling: BVALue	Sets pre-scaling offset B of the integral computation or queries the current setting.	5-116
:MATH<x>:INTEgral:RESCaling?	Queries all settings related to the rescaling of the integral computation.	5-116
:MATH<x>:INTEgral:RESCaling: AVALue	Sets rescaling coefficient A of the integral computation or queries the current setting.	5-116
:MATH<x>:INTEgral:RESCaling: BVALue	Sets rescaling offset B of the integral computation or queries the current setting.	5-116
:MATH<x>:INVert	Turns ON/OFF the inverted display of the computed waveform or queries the current setting.	5-116
:MATH<x>:IPOINT? (Initial Point)	Queries all settings related to the computation reference point.	5-117
:MATH<x>:IPOINT:JUMP	Moves the computation reference point to the specified position.	5-117
:MATH<x>:IPOINT:POSITION	Sets the computation reference point or queries the current setting.	5-117
:MATH<x>:LABel?	Queries all settings related to the label of the computed waveform.	5-117
:MATH<x>:LABel[:DEFine]	Sets the label of the computed waveform or queries the current setting.	5-117
:MATH<x>:LABel:MODE	Turns ON/OFF the label display of the computed waveform or queries the current setting.	5-117
:MATH<x>:MINus?	Queries all settings related to the subtraction.	5-117
:MATH<x>:MINus:PSCaling<x>?	Queries all settings related to the pre-scaling of the subtraction.	5-117
:MATH<x>:MINus:PSCaling<x>: AVALue	Sets pre-scaling coefficient A of the subtraction or queries the current setting.	5-117
:MATH<x>:MINus:PSCaling<x>: BVALue	Sets pre-scaling offset B of the subtraction or queries the current setting.	5-117
:MATH<x>:MINus:RESCaling?	Queries all settings related to the rescaling of the subtraction.	5-118
:MATH<x>:MINus:RESCaling:AVALue	Sets rescaling coefficient A of the subtraction or queries the current setting.	5-118
:MATH<x>:MINus:RESCaling:BVALue	Sets rescaling offset B of the subtraction or queries the current setting.	5-118
:MATH<x>:MULTiple?	Queries all settings related to the multiplication.	5-118
:MATH<x>:MULTiple:PSCaling<x>?	Queries all settings related to the pre-scaling of the multiplication.	5-118
:MATH<x>:MULTiple:PSCaling<x>: AVALue	Sets pre-scaling coefficient A of the multiplication or queries the current setting.	5-118
:MATH<x>:MULTiple:PSCaling<x>: BVALue	Sets pre-scaling offset B of the multiplication or queries the current setting.	5-118
:MATH<x>:MULTiple:RESCaling?	Queries all settings related to the rescaling of the multiplication.	5-118
:MATH<x>:MULTiple:RESCaling: AVALue	Sets rescaling coefficient A of the multiplication or queries the current setting.	5-118
:MATH<x>:MULTiple:RESCaling: BVALue	Sets rescaling offset B of the multiplication or queries the current setting.	5-119
:MATH<x>:OPERation	Sets the operator or queries the current setting.	5-119
:MATH<x>:PLUS?	Queries all settings related to the addition.	5-119
:MATH<x>:PLUS:PSCaling<x>?	Queries all settings related to the pre-scaling of the addition.	5-119
:MATH<x>:PLUS:PSCaling<x>: AVALue	Sets pre-scaling coefficient A of the addition or queries the current setting.	5-119
:MATH<x>:PLUS:PSCaling<x>: BVALue	Sets pre-scaling offset B of the addition or queries the current setting.	5-119
:MATH<x>:PLUS:RESCaling?	Queries all settings related to the rescaling of the addition.	5-119
:MATH<x>:PLUS:RESCaling:AVALue	Sets rescaling coefficient A of the addition or queries the current setting.	5-119
:MATH<x>:PLUS:RESCaling:BVALue	Sets rescaling offset B of the addition or queries the current setting.	5-119
:MATH<x>:POSITION	Sets the vertical position of the computed waveform or queries the current setting.	5-120
:MATH<x>:SCALE?	Queries all settings related to scaling.	5-120
:MATH<x>:SCALE:CENTer	Sets the offset of the computed waveform or queries the current setting.	5-120
:MATH<x>:SCALE:MODE	Sets the scaling mode or queries the current setting.	5-120
:MATH<x>:SCALE:SENSitivity	Sets the vertical sensitivity of the computed waveform or queries the current setting.	5-120
:MATH<x>:SElect	Sets the display option or queries the current setting.	5-120
:MATH<x>:SVALue (Scale VALUE)	Turns ON/OFF the scale value display or queries the current setting.	5-120

## 5.1 List of Commands

Command	Function	Page
:MATH<x>:THReshold<x>	Sets the edge detection level of the count computation or queries the current setting.	5-120
:MATH<x>:UNIT?	Queries all settings related to the computation unit.	5-120
:MATH<x>:UNIT[:DEFine]	Sets the computation unit or queries the current setting.	5-120
:MATH<x>:UNIT:MODE	Sets the automatic/manual addition of the computation unit or queries the current setting.	5-121
:MATH<x>:USERdefine?	Queries all settings related to user-defined math or queries the current setting.	5-121
:MATH<x>:USERdefine:ARAnging	Executes auto ranging for user-defined math.	5-121
:MATH<x>:USERdefine:CONStant<x>	Sets a user-defined math constant or queries the current setting.	5-121
:MATH<x>:USERdefine:DEFine	Sets a user-defined math equation or queries the current setting.	5-121
:MATH<x>:USERdefine:HISTory: ABORt	Cancels history computation for user-defined math.	5-121
:MATH<x>:USERdefine:HISTory: EXECute	Executes history computation for user-defined math.	5-121

### MEASure Group

:MEASure?	Queries all settings related to the automated measurement of waveform parameters.	5-122
:MEASure:CALCulation?	Queries all settings related to calculation items.	5-122
:MEASure:CALCulation:ALL	Turns ON/OFF all calculation items.	5-122
:MEASure:CALCulation:COUnT<x>?	Queries the statistical processing count of the calculation item.	5-122
:MEASure:CALCulation:DEFine<x>	Sets the equation of the calculation item or queries the current setting.	5-122
:MEASure:CALCulation:{MAXimum<x> > MEAN<x> MINimum<x> SDEViation <x>}?	Queries the statistical value of the calculation item.	5-122
:MEASure:CALCulation:STATe<x>	Turns ON/OFF the calculation item or queries the current setting.	5-122
:MEASure:CALCulation:VALue<x>?	Queries the automated measured value of the calculation item.	5-123
:MEASure:CONTinuous?	Queries all settings related to the continuous statistical processing.	5-123
:MEASure:CONTinuous:COUnT	Sets the continuous statistical processing count or queries the current setting.	5-123
:MEASure:CONTinuous:REStArt	Restarts the continuous statistical processing.	5-123
:MEASure:CYCLE?	Queries all settings related to the cycle statistical processing.	5-123
:MEASure:CYCLE:ABORt	Aborts the execution of the cycle statistical processing.	5-123
:MEASure:CYCLE:EXECute	Executes the cycle statistical processing. This is an overlap command.	5-123
:MEASure:CYCLE:TRACe	Sets the cycle source trace of the continuous statistical processing count or queries the current setting.	5-123
:MEASure:DISPlay	Turns ON/OFF the display of the automated measurement of waveform parameters or queries the current setting.	5-123
:MEASure:HISTory:ABORt	Aborts the execution of the statistical processing of the history data.	5-123
:MEASure:HISTory:EXECute	Executes the statistical processing of the history data. This is an overlap command.	5-123
:MEASure:MODE	Sets the mode of the automated measurement of waveform parameters or queries the current setting.	5-123
:MEASure:THReshold?	Queries all settings related to the threshold levels of the automated measurement of waveform parameters.	5-124
:MEASure:THReshold:TRACe<x>?	Queries the threshold levels of the trace.	5-124
:MEASure:THReshold:TRACe<x>: AUtO	Sets the detection mode when the auto setting of the threshold level is enabled or queries the current setting.	5-124
:MEASure:THReshold:TRACe<x>: LHYSteresis?	Queries all settings related to the level and hysteresis of the threshold level.	5-124
:MEASure:THReshold:TRACe<x>: LHYSteresis:HYSteresis	Sets the hysteresis of the threshold level or queries the current setting.	5-124
:MEASure:THReshold:TRACe<x>: LHYSteresis:LEVel	Sets the level of the threshold level or queries the current setting.	5-125
:MEASure:THReshold:TRACe<x>: MODe	Sets the setup mode of the threshold level or queries the current setting.	5-125
:MEASure:THReshold:TRACe<x>: ULOWer?	Queries all settings related to the upper and lower limits of the threshold level.	5-125
:MEASure:THReshold:TRACe<x>: ULOWer:RAnge	Sets the upper and lower limits of the threshold level or queries the current setting.	5-125
:MEASure:TRACe<x>?	Queries all settings related to the trace.	5-126
:MEASure:TRACe<x>:AREa<x>?	Queries all settings related to the area.	5-126

Command	Function	Page
:MEASure:TRACe<x>:AREA<x>:ALL	Turns ON/OFF all waveform parameters.	5-126
:MEASure:TRACe<x>: AREA<x>:<Parameter>?	Queries all settings related to the waveform parameter.	5-127
:MEASure:TRACe<x>: AREA<x>:<Parameter>:COUNT?	Queries the continuous statistical processing count of the waveform parameter.	5-127
:MEASure:TRACe<x>:AREA<x>:<Parameter>:{MAXimum MEAN MINimum SD EVIation}?	Queries the statistical value of the waveform parameter.	5-127
:MEASure:TRACe<x>: AREA<x>:<Parameter>:STATe	Turns ON/OFF the waveform parameter or queries the current setting.	5-127
:MEASure:TRACe<x>: AREA<x>:<Parameter>:VALue?	Queries the automated measured value of the waveform parameter.	5-128
:MEASure:TRACe<x>:AREA<x>: DELay:MEASure?	Queries all settings related to the measurement conditions of the source waveform of the delay measurement between channels.	5-128
:MEASure:TRACe<x>:AREA<x>: DELay:MEASure:COUNT	Sets the edge detection count of the source waveform of the delay measurement between channels or queries the current setting.	5-128
:MEASure:TRACe<x>:AREA<x>: DELay:MEASure:POLarity	Sets the polarity of the source waveform of the delay measurement between channels or queries the current setting.	5-128
:MEASure:TRACe<x>:AREA<x>: DELay:REFerence?	Queries all settings related to the reference waveform of the delay measurement between channels.	5-128
:MEASure:TRACe<x>:AREA<x>: DELay:REFerence:COUNT	Sets the edge detection count of the reference waveform of the delay measurement between channels or queries the current setting.	5-129
:MEASure:TRACe<x>:AREA<x>: DELay:REFerence:POLarity	Sets the polarity of the reference waveform of the delay measurement between channels or queries the current setting.	5-129
:MEASure:TRACe<x>:AREA<x>: DELay:REFerence:TRACe	Sets the trace of the reference waveform of the delay measurement between channels or queries the current setting.	5-129
:MEASure:TRACe<x>:AREA<x>: DELay:SOURce	Sets the reference of the delay measurement between channels or queries the current setting.	5-129
:MEASure:TRACe<x>:AREA<x>: DPRoximal?	Queries all settings related to the distal and proximal values.	5-129
:MEASure:TRACe<x>:AREA<x>: DPRoximal:MODE	Sets the unit of the distal and proximal values or queries the current setting.	5-129
:MEASure:TRACe<x>:AREA<x>: DPRoximal:PERCent	Sets the distal and proximal values as a percentage or queries the current setting.	5-130
:MEASure:TRACe<x>:AREA<x>: DPRoximal:UNIT	Sets the distal and proximal values in the specified unit or queries the current setting.	5-130
:MEASure:TRACe<x>:AREA<x>: METHod	Sets the method for detecting the High/Low level for automated measurement of waveform parameters or queries the current setting.	5-130
:MEASure:TRANge<x> (Time Range)	Sets the measurement range or queries the current setting.	5-130
:MEASure:WAIT?	Waits for the completion of the automated measurement with a timeout option.	5-130
:MEASure:WINDow<x>	Sets the measurement source window of the area or queries the current setting.	5-130

### REFerence Group

:REFerence<x>?	Queries all settings related to the reference.	5-131
:REFerence<x>:DISPlay	Turns ON/OFF the display of the reference or queries the current setting.	5-131
:REFerence<x>:INVert	Sets the inverted display of the reference or queries the current setting.	5-131
:REFerence<x>:LABel?	Queries all settings related to the waveform label of the reference.	5-131
:REFerence<x>:LABel[:DEFine]	Sets the waveform label of the reference or queries the current setting.	5-131
:REFerence<x>:LABel:MODE	Turns ON/OFF the waveform label display of the reference or queries the current setting.	5-131
:REFerence<x>:LOAD	Loads the waveform to the reference.	5-131
:REFerence<x>:POSITION	Sets the vertical position of the reference or queries the current setting.	5-131
:REFerence<x>:SELect	Sets the waveform (computation or reference) to the computation channel or queries the current setting.	5-131
:REFerence<x>:SVALue (Scale VALUE)	Turns ON/OFF the scale display of the reference or queries the current setting.	5-131

### SEARCh Group

:SEARCh<x>?	Queries all settings related to the search function.	5-132
:SEARCh<x>:ABORt	Aborts the search.	5-132
:SEARCh<x>:CLOCK?	Queries all settings related to the clock channel.	5-132

## 5.1 List of Commands

Command	Function	Page
:SEARCH<x>:CLOCK:POLarity	Sets the polarity of the clock channel or queries the current setting.	5-132
:SEARCH<x>:CLOCK:SOURce	Sets the clock trace of the search or queries the current setting.	5-132
:SEARCH<x>:DECimation	Sets the decimation detection of the skip mode or queries the current setting.	5-132
:SEARCH<x>:EXECute	Executes the search. This is an overlap command.	5-133
:SEARCH<x>:HOLDoff	Sets the hold off detection or queries the current setting.	5-133
:SEARCH<x>:LOGic	Sets the search logic or queries the current setting.	5-133
:SEARCH<x>:POLarity	Sets the search polarity or queries the current setting.	5-133
:SEARCH<x>:SElect	Sets the detection waveform number of the search function and queries the position that corresponds to the detection waveform number.	5-133
:SEARCH<x>:SElect? MAXimum	Queries the detection count of the search function.	5-133
:SEARCH<x>:SMODE	Sets the skip mode or queries the current setting.	5-133
:SEARCH<x>:SPATtern? (Serial Pattern)	Queries all settings related to the serial pattern search.	5-133
:SEARCH<x>:SPATtern:CLOCK?	Queries all settings related to clock of the serial pattern search.	5-133
:SEARCH<x>:SPATtern:CLOCK:MODE	Enables/Disables the clock of the serial pattern search or queries the current setting.	5-134
:SEARCH<x>:SPATtern:CLOCK:POLarity	Sets the polarity of the clock trace of the serial pattern search or queries the current setting.	5-134
:SEARCH<x>:SPATtern:CLOCK:SOURce	Sets the clock trace of the serial pattern search or queries the current setting.	5-134
:SEARCH<x>:SPATtern:CS	Enables/Disables the chip select of the serial pattern search or queries the current setting.	5-134
:SEARCH<x>:SPATtern:LATCH?	Queries all settings related to latch of the serial pattern search.	5-134
:SEARCH<x>:SPATtern:LATCH:POLarity	Sets the polarity of the latch trace of the serial pattern search or queries the current setting.	5-134
:SEARCH<x>:SPATtern:LATCH:TRACE	Sets the latch trace of the serial pattern search or queries the current setting.	5-134
:SEARCH<x>:SPATtern:SETup?	Queries all settings related to setup of the serial pattern search.	5-134
:SEARCH<x>:SPATtern[:SETup]:BITRate	Sets the bit rate of the serial pattern search or queries the current setting.	5-135
:SEARCH<x>:SPATtern[:SETup]:CLEar	Clears the entire pattern of the serial pattern search (to don't care).	5-135
:SEARCH<x>:SPATtern[:SETup]:DATA?	Queries all settings related to data of the serial pattern search.	5-135
:SEARCH<x>:SPATtern[:SETup]:DATA:ACTive	Sets the active level of the data trace of the serial pattern search or queries the current setting.	5-135
:SEARCH<x>:SPATtern[:SETup]:DATA:TRACE	Sets the data trace of the serial pattern search or queries the current setting.	5-135
:SEARCH<x>:SPATtern[:SETup]:HEXA	Sets the pattern of the serial pattern search in hexadecimal notation.	5-135
:SEARCH<x>:SPATtern[:SETup]:PATTern	Sets the pattern of the serial pattern search in binary notation or queries the current setting.	5-135
:SEARCH<x>:SPOint	Sets the search start position or queries the current setting.	5-135
:SEARCH<x>:STRace	Sets the search source trace or queries the current setting.	5-135
:SEARCH<x>:TRACE<x>?	Queries all settings related to the search conditions of the trace.	5-136
:SEARCH<x>:TRACE<x>:CONDition	Sets the condition to be satisfied for the trace or queries the current setting.	5-136
:SEARCH<x>:TRACE<x>:HYSTEResis	Sets the hysteresis of the trace or queries the current setting.	5-136
:SEARCH<x>:TRACE<x>:LEVel	Sets the threshold level of the trace or queries the current setting.	5-136
:SEARCH<x>:TYPE	Sets the search type or queries the current setting.	5-136
:SEARCH<x>:WIDTh?	Queries all settings related to the pulse width search.	5-136
:SEARCH<x>:WIDTh:MODE	Sets the pulse width determination mode or queries the current setting.	5-136
:SEARCH<x>:WIDTh:TIME<x>	Sets the pulse width of the pulse width search or queries the current setting.	5-136
:SEARCH<x>:WIDTh:TYPE	Sets the pulse width search type or queries the current setting.	5-136

### SNAP Group

:SNAP	Executes the snapshot.	5-137
-------	------------------------	-------

### SStart Group

:SStart?	Starts the waveform acquisition with the trigger mode set to single. If the waveform acquisition stops within the specified time period, 0 is returned at that point.	5-138
----------	---	-------

Command	Function	Page
<b>START Group</b>		
:START	Starts the waveform acquisition.	5-139
<b>STATus Group</b>		
:STATus?	Queries all settings related to the communication status function.	5-140
:STATus:CONDition?	Queries the contents of the condition register.	5-140
:STATus:EESE	Sets the extended event enable register or queries the current setting.	5-140
:STATus:EESE?	Queries the content of the extended event register and clears the register.	5-140
:STATus:ERRor?	Queries the error code and message information (top of the error queue).	5-140
:STATus:FILTer<x>	Sets the transition filter or queries the current setting.	5-140
:STATus:QENable	Sets whether to store messages other than errors to the error queue or queries the current setting.	5-140
:STATus:QMESsage	Sets whether or not to attach message information to the response to the "STATus:ERRor?" query or queries the current setting.	5-140
:STATus:SPOLL? (Serial Poll)	Executes serial polling.	5-140
<b>STOP Group</b>		
:STOP	Stops the waveform acquisition.	5-141
<b>SYSTem Group</b>		
:SYSTem?	Queries all settings related to the system.	5-142
:SYSTem:CLICk	Turns ON/OFF the click sound or queries the current setting.	5-142
:SYSTem:CLOCK?	Queries all settings related to the date, time, and time difference with respect to GMT.	5-142
:SYSTem:CLOCK:DTIME	Sets the date, time, and time difference with respect to GMT or queries the current setting.	5-142
:SYSTem:CLOCK:MODE	Turns ON/OFF the date, time, and time difference with respect to GMT or queries the current setting.	5-142
:SYSTem:FORMat:IMEMory[:EXECute]	Formats the internal memory.	5-142
:SYSTem:FORMat:IHDD[:EXECute]	Formats the internal hard disk.	5-142
:SYSTem:FORMat:SDElete[:EXECute] (Sure Delete)	Clears and formats the internal memory.	5-142
:SYSTem:LANGuage	Sets the message language or queries the current setting.	5-142
:SYSTem:MFSize	Sets the menu font size or queries the current setting.	5-142
:SYSTem:MLANGuage	Sets the menu language or queries the current setting.	5-142
:SYSTem:OVERview	Displays system information.	5-142
:SYSTem:USBKeyboard	Sets the USB keyboard type or queries the current setting.	5-142
<b>TELEcomtest Group</b>		
:TELEcomtest?	Queries all settings related to the telecom test.	5-143
:TELEcomtest:CATegory	Sets the telecom test type or queries the current setting.	5-143
:TELEcomtest:DISPlay	Turns ON/OFF the telecom test display or queries the current setting.	5-143
:TELEcomtest:EYEPattern?	Queries all settings related to the eye pattern.	5-143
:TELEcomtest:EYEPattern:ALL	Turns ON/OFF all eye pattern parameters.	5-143
:TELEcomtest:EYEPattern:<Parameter>?	Queries all settings related to the waveform parameter of the eye pattern.	5-143
:TELEcomtest:EYEPattern:<Parameter>:STATe	Turns ON/OFF the waveform parameter of the eye pattern or queries the current setting.	5-144
:TELEcomtest:EYEPattern:<Parameter>:VALue?	Queries the waveform parameter value of the eye pattern.	5-144
:TELEcomtest:EYEPattern:TLEVels?	Queries all settings related to the threshold level of the eye pattern.	5-144
:TELEcomtest:EYEPattern:TLEVels:MODE	Sets the unit of the threshold level of the eye pattern or queries the current setting.	5-144
:TELEcomtest:EYEPattern:TLEVels:PERCent	Sets the threshold level of the eye pattern as a percentage or queries the current setting.	5-144
:TELEcomtest:EYEPattern:TLEVels:UNIT	Sets the threshold level of the eye pattern in UNIT or queries the current setting.	5-144
:TELEcomtest:EYEPattern:VDARK	Sets the dark level (zero light level) or queries the current setting.	5-144

## 5.1 List of Commands

Command	Function	Page
:TELecomtest:MASK?	Queries all settings related to the mask test.	5-145
:TELecomtest:MASK:ELEment<x>?	Queries all settings related to the element used in the mask test.	5-145
:TELecomtest:MASK:ELEment<x>: ALL	Turns ON/OFF all items of the element.	5-145
:TELecomtest:MASK:ELEment<x>: PSPCount? (Sample Point Count %)	Queries the settings related to the error rate for the number of sampled data points of the element.	5-145
:TELecomtest:MASK:ELEment<x>: PSPCount:STATe	Turns ON/OFF the measurement of the error rate for the number of sampled data points of the element or queries the current setting.	5-145
:TELecomtest:MASK:ELEment<x>: PSPCount:VALue?	Queries the error rate for the number of sampled data points of the element.	5-145
:TELecomtest:MASK:ELEment<x>: PWCCount? (Wave Count %)	Queries the settings related to the error rate for the acquisition count of the element.	5-145
:TELecomtest:MASK:ELEment<x>: PWCCount:STATe	Turns ON/OFF the measurement of the error rate for the acquisition count of the element or queries the current setting.	5-146
:TELecomtest:MASK:ELEment<x>: PWCCount:VALue?	Queries the error rate for the acquisition count of the element.	5-146
:TELecomtest:MASK:ELEment<x>: SPCCount? (Sample Point Count)	Queries the settings related to the number of sampled data points for the element that results in error.	5-146
:TELecomtest:MASK:ELEment<x>: SPCCount:STATe	Turns ON/OFF the measurement of the number of sampled data points for the element that results in error or queries the current setting.	5-146
:TELecomtest:MASK:ELEment<x>: SPCCount:VALue?	Queries the number of sampled data points for the element that resulted in error.	5-146
:TELecomtest:MASK:ELEment<x>: WCOunt?(Wave Count)	Queries the settings related to the acquisition count for the element that results in error.	5-146
:TELecomtest:MASK:ELEment<x>: WCOunt:STATe	Turns ON/OFF the measurement of the acquisition count for the element that results in error or queries the current setting.	5-146
:TELecomtest:MASK:ELEment<x>: WCOunt:VALue?	Queries the acquisition count for the element that resulted in error.	5-146
:TELecomtest:MMODE	Turns ON/OFF the computed waveform or queries the current setting.	5-147
:TELecomtest:TRACe	Sets the source trace of the telecom test or queries the current setting.	5-147
:TELecomtest:TRANge (Time Range)	Sets the measurement range of the telecom test or queries the current setting.	5-147
:TELecomtest:WINDow	Sets the measurement source window of the telecom test or queries the current setting.	5-147

### TIMEbase Group

:TIMEbase?	Queries all settings related to the time base.	5-148
:TIMEbase:SRATE? (Sample RATE)	Queries the sample rate or queries the current setting.	5-148
:TIMEbase:TDIV	Sets the T/div value or queries the current setting.	5-148

### TRIGger Group

:TRIGger?	Queries all settings related to the trigger.	5-149
:TRIGger:ACTion?	Queries all settings related to action-on-trigger.	5-149
:TRIGger:ACTion:ACQCount	Sets the action count of action-on-trigger or queries the current setting.	5-149
:TRIGger:ACTion:BUZZer	Sets whether to sound a buzzer when an action is activated or queries the current setting.	5-149
:TRIGger:ACTion:HCOPY	Sets whether or not to output screen image data (ON/OFF) when an action is activated, or queries the current setting.	5-149
:TRIGger:ACTion:MAIL?	Queries all settings related to the mail transmission when an action is activated.	5-149
:TRIGger:ACTion:MAIL:INTerval	Sets the interval at which to send mail when an action is activated or queries the current setting.	5-150
:TRIGger:ACTion:MAIL:MODE	Sets whether to send mail when an action is activated or queries the current setting.	5-150
:TRIGger:ACTion:MODE	Sets the action-on-trigger mode or queries the current setting.	5-150
:TRIGger:ACTion:SAVE	Sets whether to save the waveform data to the storage medium (ON/OFF) when an action is activated or queries the current setting.	5-150
:TRIGger:ACTion:STARt	Starts the action-on-trigger.	5-150
:TRIGger:ACTion:STOP	Stops the action-on-trigger.	5-150
:TRIGger:CLOCK?	Queries all settings related to the clock channel.	5-150
:TRIGger:CLOCK:POLarity	Sets the polarity of the clock channel or queries the current setting.	5-150
:TRIGger:CLOCK:SOURce	Sets the source waveform of the clock channel or queries the current setting.	5-150
:TRIGger:DELay?	Queries all settings related to the trigger delay.	5-150



## 5.1 List of Commands

Command	Function	Page
:TRIGger:DElay:EDGEcount?	Queries all settings related to edge count of the trigger delay.	5-150
:TRIGger:DElay:EDGEcount:COUNT	Sets the edge count value of the trigger delay or queries the current setting.	5-150
:TRIGger:DElay:MODE	Turns ON/OFF the trigger delay or queries the current setting.	5-151
:TRIGger:DElay:POLarity	Sets the edge polarity the trigger delay or queries the current setting.	5-151
:TRIGger:DElay:SOURce	Sets the edge source the trigger delay or queries the current setting.	5-151
:TRIGger:DElay:TIME	Sets the delay value the trigger delay or queries the current setting.	5-151
:TRIGger:DElay:TYPE	Sets the trigger delay type or queries the current setting.	5-151
:TRIGger:EINterval?	Queries all settings related to the event interval.	5-151
:TRIGger:EINterval:EVENT<x>?	Queries all settings related to the event.	5-152
:TRIGger:EINterval:EVENT<x>:CANBus?	Queries all settings related to the CAN bus signal trigger of the event.	5-152
:TRIGger:EINterval:EVENT<x>:CANBus:ACK	Sets the ACK condition of the CAN bus signal trigger or queries the current setting.	5-153
:TRIGger:EINterval:EVENT<x>:CANBus:BRATe	Sets the bit rate (data transfer rate) of the CAN bus signal trigger or queries the current setting.	5-153
:TRIGger:EINterval:EVENT<x>:CANBus:DATA?	Queries all settings related to the CAN bus signal trigger data.	5-153
:TRIGger:EINterval:EVENT<x>:CANBus:DATA:BORDer	Sets the byte order of the CAN bus signal trigger data or queries the current setting.	5-153
:TRIGger:EINterval:EVENT<x>:CANBus:DATA:CONDition	Sets the data condition of the CAN bus signal trigger or queries the current setting.	5-153
:TRIGger:EINterval:EVENT<x>:CANBus:DATA:DATA<x>	Sets the comparison data of the CAN bus signal trigger data or queries the current setting.	5-154
:TRIGger:EINterval:EVENT<x>:CANBus:DATA:DLC	Sets the number of valid bytes (DLC) of the CAN bus signal trigger data or queries the current setting.	5-154
:TRIGger:EINterval:EVENT<x>:CANBus:DATA:HEXA	Sets the CAN bus signal trigger data in hexadecimal notation.	5-154
:TRIGger:EINterval:EVENT<x>:CANBus:DATA:MSBLSb	Sets the MSB and LSB bits of the CAN bus signal trigger data or queries the current setting.	5-154
:TRIGger:EINterval:EVENT<x>:CANBus:DATA:PATtern	Sets the CAN bus signal trigger data in binary notation or queries the current setting.	5-154
:TRIGger:EINterval:EVENT<x>:CANBus:DATA:SIGN	Sets the sign of the CAN bus signal trigger data or queries the current setting.	5-155
:TRIGger:EINterval:EVENT<x>:CANBus:IDEXt?	Queries all settings related to the ID of the extended format of the CAN bus signal trigger.	5-155
:TRIGger:EINterval:EVENT<x>:CANBus:IDEXt:HEXA	Sets the ID of the extended format of the CAN bus signal trigger in hexadecimal notation.	5-155
:TRIGger:EINterval:EVENT<x>:CANBus:IDEXt:PATtern	Sets the ID of the extended format of the CAN bus signal trigger in binary notation or queries the current setting.	5-155
:TRIGger:EINterval:EVENT<x>:CANBus:IDOR?	Queries all settings related to the OR condition of the CAN bus signal trigger.	5-156
:TRIGger:EINterval:EVENT<x>:CANBus:IDOR:ID<x>?	Queries all settings related to each ID of the OR condition of the CAN bus signal trigger.	5-156
:TRIGger:EINterval:EVENT<x>:CANBus:IDOR:ID<x>:ACK	Sets each ACK condition of the OR condition of the CAN bus signal trigger or queries the current setting.	5-156
:TRIGger:EINterval:EVENT<x>:CANBus:IDOR:ID<x>:DATA?	Queries all settings related to each data of the OR condition of the CAN bus signal trigger.	5-157
:TRIGger:EINterval:EVENT<x>:CANBus:IDOR:ID<x>:DATA:BORDer	Sets byte order of each data of the OR condition of the CAN bus signal trigger or queries the current setting.	5-157
:TRIGger:EINterval:EVENT<x>:CANBus:IDOR:ID<x>:DATA:CONDition	Sets each data condition of the OR condition of the CAN bus signal trigger or queries the current setting.	5-157
:TRIGger:EINterval:EVENT<x>:CANBus:IDOR:ID<x>:DATA:DATA<x>	Sets comparison data of each data of the OR condition of the CAN bus signal trigger or queries the current setting.	5-157
:TRIGger:EINterval:EVENT<x>:CANBus:IDOR:ID<x>:DATA:DLC	Sets the number of valid bytes (DLC) of each data of the OR condition of the CAN bus signal trigger or queries the current setting.	5-157
:TRIGger:EINterval:EVENT<x>:CANBus:IDOR:ID<x>:DATA:HEXA	Sets each data of the OR condition of the CAN bus signal trigger in hexadecimal notation.	5-158
:TRIGger:EINterval:EVENT<x>:CANBus:IDOR:ID<x>:DATA:MSBLSb	Sets the MSB and LSB bits of each data of the OR condition of the CAN bus signal trigger or queries the current setting.	5-158
:TRIGger:EINterval:EVENT<x>:CANBus:IDOR:ID<x>:DATA:PATtern	Sets each data of the OR condition of the CAN bus signal trigger in binary notation or queries the current setting.	5-158
:TRIGger:EINterval:EVENT<x>:CANBus:IDOR:ID<x>:DATA:SIGN	Sets sign of each data of the OR condition of the CAN bus signal trigger or queries the current setting.	5-158

## 5.1 List of Commands

Command	Function	Page
:TRIGger:EINterval:EVENT<x>: CANBus:IDOR:ID<x>:FORmat	Sets each message format (standard or extended) of the OR condition of the CAN bus signal trigger or queries the current setting.	5-158
:TRIGger:EINterval:EVENT<x>: CANBus:IDOR:ID<x>:IDExt?	Queries all settings related to the ID of each extended format of the OR condition of the CAN bus signal trigger.	5-158
:TRIGger:EINterval:EVENT<x>: CANBus:IDOR:ID<x>:IDExt:HEXA	Sets the ID of each extended format of the OR condition of the CAN bus signal trigger in hexadecimal notation.	5-159
:TRIGger:EINterval:EVENT<x>: CANBus:IDOR:ID<x>:IDExt:PATtern	Sets the ID of each extended format of the OR condition of the CAN bus signal trigger in binary notation or queries the current setting.	5-159
:TRIGger:EINterval:EVENT<x>: CANBus:IDOR:ID<x>:IDSTd?	Queries all settings related to the ID of each standard format of the OR condition of the CAN bus signal trigger.	5-159
:TRIGger:EINterval:EVENT<x>: CANBus:IDOR:ID<x>:IDSTd:HEXA	Sets the ID of each standard format of the OR condition of the CAN bus signal trigger in hexadecimal notation.	5-159
:TRIGger:EINterval:EVENT<x>: CANBus:IDOR:ID<x>:IDSTd:PATtern	Sets the ID of each standard format of the OR condition of the CAN bus signal trigger in binary notation or queries the current setting.	5-159
:TRIGger:EINterval:EVENT<x>: CANBus:IDOR:ID<x>:MODE	Enables or disables each condition of the OR condition of the CAN bus signal trigger or queries the current setting.	5-159
:TRIGger:EINterval:EVENT<x>: CANBus:IDOR:ID<x>:RTR	Sets each RTR of the OR condition of the CAN bus signal trigger or queries the current setting.	5-160
:TRIGger:EINterval:EVENT<x>: CANBus:IDSTd?	Queries all settings related to the ID of the standard format of the CAN bus signal trigger.	5-160
:TRIGger:EINterval:EVENT<x>: CANBus:IDSTd:HEXA	Sets the ID of the standard format of the CAN bus signal trigger in hexadecimal notation.	5-160
:TRIGger:EINterval:EVENT<x>: CANBus:IDSTd:PATtern	Sets the ID of the standard format of the CAN bus signal trigger in binary notation or queries the current setting.	5-160
:TRIGger:EINterval:EVENT<x>: CANBus:MODE	Sets the CAN bus signal trigger mode or queries the current setting.	5-160
:TRIGger:EINterval:EVENT<x>: CANBus:REcEssive	Sets the recessive level (bus level) of the CAN bus signal trigger or queries the current setting.	5-160
:TRIGger:EINterval:EVENT<x>: CANBus:RTR	Sets the RTR of the CAN bus signal trigger or queries the current setting.	5-161
:TRIGger:EINterval:EVENT<x>: CANBus:SOURce	Sets the trigger source of the CAN bus signal trigger or queries the current setting.	5-161
:TRIGger:EINterval:EVENT<x>: CANBus:SPOint	Sets the sample point of the CAN bus signal trigger or queries the current setting.	5-161
:TRIGger:EINterval:EVENT<x>: CLOCK?	Queries all settings related to the clock channel of the event.	5-161
:TRIGger:EINterval:EVENT<x>: CLOCK:POLarity	Sets the polarity of the clock channel of the event or queries the current setting.	5-161
:TRIGger:EINterval:EVENT<x>: CLOCK:SOURce	Sets the source waveform of the clock channel of the event or queries the current setting.	5-161
:TRIGger:EINterval:EVENT<x>: ESTate?	Queries all settings related to the edge/state trigger.	5-161
:TRIGger:EINterval:EVENT<x>: ESTate:POLarity	Sets the polarity of the edge/state trigger or queries the current setting.	5-162
:TRIGger:EINterval:EVENT<x>: ESTate:SOURce	Sets the trigger source of the edge/state trigger or queries the current setting.	5-162
:TRIGger:EINterval:EVENT<x>: I2CBus?	Queries all settings related to the I <sup>2</sup> C bus trigger of the event.	5-162
:TRIGger:EINterval:EVENT<x>: I2CBus:ADATa?	Queries all settings related to the address of the I <sup>2</sup> C bus trigger.	5-162
:TRIGger:EINterval:EVENT<x>: I2CBus:ADATa:BIT10address?	Queries all settings related to the 10-bit address of the I <sup>2</sup> C bus trigger.	5-163
:TRIGger:EINterval:EVENT<x>: I2CBus:ADATa:BIT10address:HEXA	Sets the 10-bit address of the I <sup>2</sup> C bus trigger in hexadecimal notation.	5-163
:TRIGger:EINterval:EVENT<x>: I2CBus:ADATa:BIT10address: PATtern	Sets the 10-bit address of the I <sup>2</sup> C bus trigger in binary notation or queries the current setting.	5-163
:TRIGger:EINterval:EVENT<x>: I2CBus:ADATa:BIT7Address?	Queries all settings related to the 7-bit address of the I <sup>2</sup> C bus trigger.	5-163
:TRIGger:EINterval:EVENT<x>: I2CBus:ADATa:BIT7Address:HEXA	Sets the 7-bit address of the I <sup>2</sup> C bus trigger in hexadecimal notation.	5-163
:TRIGger:EINterval:EVENT<x>: I2CBus:ADATa:BIT7Address: PATtern	Sets the 7-bit address of the I <sup>2</sup> C bus trigger in binary notation or queries the current setting.	5-163

## 5.1 List of Commands

Command	Function	Page
:TRIGger:EINTerval:EVENT<x>: I2Cbus:ADATa:BIT7APsub?	Queries all settings related to the 7-bit + Sub address of the I <sup>2</sup> C bus trigger.	5-163
:TRIGger:EINTerval:EVENT<x>: I2Cbus:ADATa:BIT7APsub:ADDRess?	Queries all settings related to the 7-bit address of the 7-bit + Sub address of the I <sup>2</sup> C bus trigger.	5-164
:TRIGger:EINTerval:EVENT<x>: I2Cbus:ADATa:BIT7APsub:ADDRess: HEXA	Sets the 7-bit address of the 7-bit + Sub address of the I <sup>2</sup> C bus trigger in hexadecimal notation.	5-164
:TRIGger:EINTerval:EVENT<x>: I2Cbus:ADATa:BIT7APsub:ADDRess: PATTern	Sets the 7-bit address of the 7-bit + Sub address of the I <sup>2</sup> C bus trigger in binary notation or queries the current setting.	5-164
:TRIGger:EINTerval:EVENT<x>: I2Cbus:ADATa:BIT7APsub: SADDRess?	Queries all settings related to the Sub address of the 7-bit + Sub address of the I <sup>2</sup> C bus trigger.	5-164
:TRIGger:EINTerval:EVENT<x>: I2Cbus:ADATa:BIT7APsub: SADDRess:HEXA	Sets the Sub address of the 7-bit + Sub address of the I <sup>2</sup> C bus trigger in hexadecimal notation.	5-164
:TRIGger:EINTerval:EVENT<x>: I2Cbus:ADATa:BIT7APsub: SADDRess:PATTern	Sets the Sub address of the 7-bit + Sub address of the I <sup>2</sup> C bus trigger in binary notation or queries the current setting.	5-164
:TRIGger:EINTerval:EVENT<x>: I2Cbus:ADATa:TYPE	Sets the address type of the I <sup>2</sup> C bus trigger or queries the current setting.	5-164
:TRIGger:EINTerval:EVENT<x>: I2Cbus:CLOCK?	Queries all settings related to the clock of the I <sup>2</sup> C bus trigger.	5-165
:TRIGger:EINTerval:EVENT<x>: I2Cbus:CLOCK:SOURce	Sets the clock trace of the I <sup>2</sup> C bus trigger or queries the current setting.	5-165
:TRIGger:EINTerval:EVENT<x>: I2Cbus:DATA?	Queries all settings related to the data of the I <sup>2</sup> C bus trigger.	5-165
:TRIGger:EINTerval:EVENT<x>: I2Cbus:DATA:BYTE	Sets the number of data bytes of the I <sup>2</sup> C bus trigger or queries the current setting.	5-165
:TRIGger:EINTerval:EVENT<x>: I2Cbus:DATA:CONDition	Sets the determination method (match or not match) of the data of the I <sup>2</sup> C bus trigger or queries the current setting.	5-165
:TRIGger:EINTerval:EVENT<x>: I2Cbus:DATA:DPOSition	Sets the position for comparing the data pattern of the I <sup>2</sup> C bus trigger or queries the current setting.	5-165
:TRIGger:EINTerval:EVENT<x>: I2Cbus:DATA:HEXA<x>	Sets the data of the I <sup>2</sup> C bus trigger in hexadecimal notation.	5-166
:TRIGger:EINTerval:EVENT<x>: I2Cbus:DATA:MODE	Enables/Disables the data conditions of the I <sup>2</sup> C bus trigger or queries the current setting.	5-166
:TRIGger:EINTerval:EVENT<x>: I2Cbus:DATA:PATTern<x>	Sets the data of the I <sup>2</sup> C bus trigger in binary notation or queries the current setting.	5-166
:TRIGger:EINTerval:EVENT<x>: I2Cbus:DATA:PMODE	Sets the pattern comparison start position mode of the data of the I <sup>2</sup> C bus trigger or queries the current setting.	5-166
:TRIGger:EINTerval:EVENT<x>: I2Cbus:DATA:SOURce	Sets the data trace of the I <sup>2</sup> C bus trigger or queries the current setting.	5-166
:TRIGger:EINTerval:EVENT<x>: I2Cbus:GCALL?	Queries all settings related to the general call of the I <sup>2</sup> C bus trigger.	5-166
:TRIGger:EINTerval:EVENT<x>: I2Cbus:GCALL:BIT7maddress?	Queries all settings related to the 7-bit master address of the general call of the I <sup>2</sup> C bus trigger.	5-167
:TRIGger:EINTerval:EVENT<x>: I2Cbus:GCALL:BIT7maddress:HEXA	Sets the 7-bit master address of the general call of the I <sup>2</sup> C bus trigger in hexadecimal notation.	5-167
:TRIGger:EINTerval:EVENT<x>: I2Cbus:GCALL:BIT7maddress: PATTern	Sets the 7-bit master address of the general call of the I <sup>2</sup> C bus trigger in binary notation or queries the current setting.	5-167
:TRIGger:EINTerval:EVENT<x>: I2Cbus:GCALL:SBYTE (Second Byte)	Sets the second byte type of the general call of the I <sup>2</sup> C bus trigger or queries the current setting.	5-167
:TRIGger:EINTerval:EVENT<x>: I2Cbus:MODE	Sets the trigger mode of the I <sup>2</sup> C bus trigger or queries the current setting.	5-167
:TRIGger:EINTerval:EVENT<x>: I2Cbus:NAIGNore?	Queries all settings related to the NON ACK ignore mode of the I <sup>2</sup> C bus trigger.	5-167
:TRIGger:EINTerval:EVENT<x>: I2Cbus:NAIGNore:HSMODE	Sets whether to ignore NON ACK in high speed mode of the I <sup>2</sup> C bus trigger or queries the current setting.	5-168
:TRIGger:EINTerval:EVENT<x>: I2Cbus:NAIGNore:RACcess	Sets whether to ignore NON ACK in read access mode of the I <sup>2</sup> C bus trigger or queries the current setting.	5-168
:TRIGger:EINTerval:EVENT<x>: I2Cbus:NAIGNore:SBYTE (Start Byte)	Sets whether to ignore NON ACK in the start byte of the I <sup>2</sup> C bus trigger or queries the current setting.	5-168

## 5.1 List of Commands

Command	Function	Page
:TRIGger:EINTerval:EVENT<x>: I2CBus:SBHSMODE?	Queries all settings related to the start byte and high speed mode of the I <sup>2</sup> C bus trigger.	5-168
:TRIGger:EINTerval:EVENT<x>: I2CBus:SBHSMODE:TYPE	Sets the type of the start byte or high speed mode of the I <sup>2</sup> C bus trigger or queries the current setting.	5-168
:TRIGger:EINTerval:EVENT<x>: LINBus?	Queries all settings related to LIN bus signal triggers of each event.	5-168
:TRIGger:EINTerval:EVENT<x>: LINBus:BRATE	Sets the LIN bus signal trigger bitrate (data transfer rate) or queries the current setting.	5-168
:TRIGger:EINTerval:EVENT<x>: LINBus:SOURCE	Sets the LIN bus signal trigger source or queries the current setting.	5-169
:TRIGger:EINTerval:EVENT<x>: SPATtern? (Serial Pattern)	Queries all settings related to the serial pattern trigger of the event.	5-169
:TRIGger:EINTerval:EVENT<x>: SPATtern:BITRate	Sets the bit rate of the serial pattern trigger or queries the current setting.	5-169
:TRIGger:EINTerval:EVENT<x>: SPATtern:CLEAR	Clears the entire pattern of the serial pattern trigger (to don't care).	5-169
:TRIGger:EINTerval:EVENT<x>: SPATtern:CLOCK?	Queries all settings related to clock of the serial pattern trigger.	5-169
:TRIGger:EINTerval:EVENT<x>: SPATtern:CLOCK:MODE	Enables/Disables the clock of the serial pattern trigger or queries the current setting.	5-169
:TRIGger:EINTerval:EVENT<x>: SPATtern:CLOCK:POLarity	Sets the polarity of the clock trace of the serial pattern trigger or queries the current setting.	5-169
:TRIGger:EINTerval:EVENT<x>: SPATtern:CLOCK:SOURCE	Sets the clock trace of the serial pattern trigger or queries the current setting.	5-170
:TRIGger:EINTerval:EVENT<x>: SPATtern:CS	Enables/Disables the chip select of the serial pattern trigger or queries the current setting.	5-170
:TRIGger:EINTerval:EVENT<x>: SPATtern:DATA?	Queries all settings related to data of the serial pattern trigger.	5-170
:TRIGger:EINTerval:EVENT<x>: SPATtern:DATA:ACTIVE	Sets the active level of the data of the serial pattern trigger or queries the current setting.	5-170
:TRIGger:EINTerval:EVENT<x>: SPATtern:DATA:SOURCE	Sets the data trace of the serial pattern trigger or queries the current setting.	5-170
:TRIGger:EINTerval:EVENT<x>: SPATtern:HEXA	Sets the pattern of the serial pattern trigger in hexadecimal notation.	5-170
:TRIGger:EINTerval:EVENT<x>: SPATtern:LATCH?	Queries all settings related to latch of the serial pattern trigger.	5-170
:TRIGger:EINTerval:EVENT<x>: SPATtern:LATCH:POLarity	Sets the polarity of the latch trace of the serial pattern trigger or queries the current setting.	5-171
:TRIGger:EINTerval:EVENT<x>: SPATtern:LATCH:SOURCE	Sets the latch trace of the serial pattern trigger or queries the current setting.	5-171
:TRIGger:EINTerval:EVENT<x>: SPATtern:PATTERN	Sets the pattern of the serial pattern trigger in binary notation or queries the current setting.	5-171
:TRIGger:EINTerval:EVENT<x>: SPIBus?	Queries all settings related to the SPI bus trigger of the event.	5-171
:TRIGger:EINTerval:EVENT<x>: SPIBus:BITOrder	Sets the bit order of the SPI bus trigger or queries the current setting.	5-171
:TRIGger:EINTerval:EVENT<x>: SPIBus:CLOCK?	Queries all settings related to the clock of the SPI bus trigger.	5-172
:TRIGger:EINTerval:EVENT<x>: SPIBus:CLOCK:POLarity	Sets the polarity of the clock trace of the SPI bus trigger or queries the current setting.	5-172
:TRIGger:EINTerval:EVENT<x>: SPIBus:CLOCK:SOURCE	Sets the clock trace of the SPI bus trigger or queries the current setting.	5-172
:TRIGger:EINTerval:EVENT<x>: SPIBus:CS?	Queries all settings related to the chip select of the SPI bus trigger.	5-172
:TRIGger:EINTerval:EVENT<x>: SPIBus:CS:ACTIVE	Sets the active level of the chip select of the SPI bus trigger or queries the current setting.	5-172
:TRIGger:EINTerval:EVENT<x>: SPIBus:CS:SOURCE	Sets the chip select trace of the SPI bus trigger or queries the current setting.	5-172
:TRIGger:EINTerval:EVENT<x>: SPIBus:DATA<x>?	Queries all settings related to the data of the SPI bus trigger.	5-172
:TRIGger:EINTerval:EVENT<x>: SPIBus:DATA<x>:BYTE	Sets the number of bytes of the data of the SPI bus trigger or queries the current setting.	5-173
:TRIGger:EINTerval:EVENT<x>: SPIBus:DATA<x>:CONDition	Sets the determination method (match or not match) of the data of the SPI bus trigger or queries the current setting.	5-173

## 5.1 List of Commands

Command	Function	Page
:TRIGger:EINterval:EVENT<x>: SPIBus:DATA<x>:DPOsition	Sets the pattern comparison start position of the data of the SPI bus trigger or queries the current setting.	5-173
:TRIGger:EINterval:EVENT<x>: SPIBus:DATA<x>:HEXA<x>	Sets the data of the SPI bus trigger in hexadecimal notation.	5-173
:TRIGger:EINterval:EVENT<x>: SPIBus:DATA<x>:PATtern<x>	Sets the data of the SPI bus trigger in binary notation or queries the current setting.	5-173
:TRIGger:EINterval:EVENT<x>: SPIBus:DATA<x>:SOURce	Sets the trace of the data of the SPI bus trigger or queries the current setting.	5-173
:TRIGger:EINterval:EVENT<x>: SPIBus:MODE	Sets the wiring system of the SPI bus trigger (three-wire or four-wire) or queries the current setting.	5-174
:TRIGger:EINterval:EVENT<x>: STATe?	Queries all settings related to condition to be satisfied of the event.	5-174
:TRIGger:EINterval:EVENT<x>: STATe:CHANnel<x>	Sets the condition to be satisfied of the channel or queries the current setting.	5-174
:TRIGger:EINterval:EVENT<x>: STATe:LOGic	Sets the logic of the condition to be satisfied or queries the current setting.	5-174
:TRIGger:EINterval:EVENT<x>: TYPE	Sets the trigger type of the event or queries the current setting.	5-174
:TRIGger:EINterval:EVENT<x>: WIDTh?	Queries all settings related to the pulse width trigger of the event.	5-174
:TRIGger:EINterval:EVENT<x>: WIDTh:MODE	Sets the determination mode of the pulse width trigger or queries the current setting.	5-175
:TRIGger:EINterval:EVENT<x>: WIDTh:POLarity	Sets the polarity of the pulse width trigger or queries the current setting.	5-175
:TRIGger:EINterval:EVENT<x>: WIDTh:SOURce	Sets the trigger source of the pulse width trigger or queries the current setting.	5-175
:TRIGger:EINterval:EVENT<x>: WIDTh:TIME<x>	Sets the pulse width of the pulse width trigger or queries the current setting.	5-175
:TRIGger:EINterval:MODE	Sets the determination mode of the event interval or queries the current setting.	5-175
:TRIGger:EINterval:TIME<x>	Sets the interval time of the event interval or queries the current setting.	5-176
:TRIGger:EINterval:TRY?	Queries all settings related to the event interval trial.	5-176
:TRIGger:EINterval:TRY:MODE	Sets the trial mode or queries the current setting.	5-176
:TRIGger:EINterval:TRY:SElect	Sets the source event of the trial mode or queries the current setting.	5-176
:TRIGger:ENHanced?	Queries all settings related to the enhanced trigger.	5-176
:TRIGger:ENHanced:CANBus?	Queries all settings related to the CAN bus signal trigger.	5-177
:TRIGger:ENHanced:CANBus:ACK	Sets the ACK condition of the CAN bus signal trigger or queries the current setting.	5-177
:TRIGger:ENHanced:CANBus:BRATe	Sets the bit rate (data transfer rate) of the CAN bus signal trigger or queries the current setting.	5-177
:TRIGger:ENHanced:CANBus:DATA?	Queries all settings related to the CAN bus signal trigger data.	5-177
:TRIGger:ENHanced:CANBus:DATA: BORDER	Sets the byte order of the CAN bus signal trigger data or queries the current setting.	5-177
:TRIGger:ENHanced:CANBus:DATA: CONDition	Sets the data condition of the CAN bus signal trigger or queries the current setting.	5-177
:TRIGger:ENHanced:CANBus:DATA: DATA<x>	Sets the comparison data of the CAN bus signal trigger data or queries the current setting.	5-178
:TRIGger:ENHanced:CANBus:DATA: DLC	Sets the number of valid bytes (DLC) of the CAN bus signal trigger data or queries the current setting.	5-178
:TRIGger:ENHanced:CANBus:DATA: HEXA	Sets the CAN bus signal trigger data in hexadecimal notation.	5-178
:TRIGger:ENHanced:CANBus:DATA: MSBLsb	Sets the MSB and LSB bits of the CAN bus signal trigger data or queries the current setting.	5-178
:TRIGger:ENHanced:CANBus:DATA: PATtern	Sets the CAN bus signal trigger data in binary notation or queries the current setting.	5-178
:TRIGger:ENHanced:CANBus:DATA: SIGN	Sets the sign of the CAN bus signal trigger data or queries the current setting.	5-178
:TRIGger:ENHanced:CANBus:IDExt?	Queries all settings related to the ID of the extended format of the CAN bus signal trigger.	5-178
:TRIGger:ENHanced:CANBus:IDExt: HEXA	Sets the ID of the extended format of the CAN bus signal trigger in hexadecimal notation.	5-179
:TRIGger:ENHanced:CANBus:IDExt: PATtern	Sets the ID of the extended format of the CAN bus signal trigger in binary notation or queries the current setting.	5-179
:TRIGger:ENHanced:CANBus:IDOR?	Queries all settings related to the OR condition of the CAN bus signal trigger.	5-179

## 5.1 List of Commands

Command	Function	Page
:TRIGger:ENHanced:CANBus:IDOR:ID<x>?	Queries all settings related to each ID of the OR condition of the CAN bus signal trigger.	5-179
:TRIGger:ENHanced:CANBus:IDOR:ID<x>:ACK	Sets each ACK condition of the OR condition of the CAN bus signal trigger or queries the current setting.	5-180
:TRIGger:ENHanced:CANBus:IDOR:ID<x>:DATA?	Queries all settings related to each data of the OR condition of the CAN bus signal trigger.	5-180
:TRIGger:ENHanced:CANBus:IDOR:ID<x>:DATA:BORDER	Sets byte order of each data of the OR condition of the CAN bus signal trigger or queries the current setting.	5-180
:TRIGger:ENHanced:CANBus:IDOR:ID<x>:DATA:CONDition	Sets each data condition of the OR condition of the CAN bus signal trigger or queries the current setting.	5-180
:TRIGger:ENHanced:CANBus:IDOR:ID<x>:DATA:DATA<x>	Sets comparison data of each data of the OR condition of the CAN bus signal trigger or queries the current setting.	5-180
:TRIGger:ENHanced:CANBus:IDOR:ID<x>:DATA:DLC	Sets the number of valid bytes (DLC) of each data of the OR condition of the CAN bus signal trigger or queries the current setting.	5-181
:TRIGger:ENHanced:CANBus:IDOR:ID<x>:DATA:HEXA	Sets each data of the OR condition of the CAN bus signal trigger in hexadecimal notation.	5-181
:TRIGger:ENHanced:CANBus:IDOR:ID<x>:DATA:MSBLSb	Sets the MSB and LSB bits of each data of the OR condition of the CAN bus signal trigger or queries the current setting.	5-181
:TRIGger:ENHanced:CANBus:IDOR:ID<x>:DATA:PATtern	Sets each data of the OR condition of the CAN bus signal trigger in binary notation or queries the current setting.	5-181
:TRIGger:ENHanced:CANBus:IDOR:ID<x>:DATA:SIGN	Sets sign of each data of the OR condition of the CAN bus signal trigger or queries the current setting.	5-181
:TRIGger:ENHanced:CANBus:IDOR:ID<x>:FORMat	Sets each message format (standard or extended) of the OR condition of the CAN bus signal trigger or queries the current setting.	5-181
:TRIGger:ENHanced:CANBus:IDOR:ID<x>:IDEXt?	Queries all settings related to the ID of each extended format of the OR condition of the CAN bus signal trigger.	5-182
:TRIGger:ENHanced:CANBus:IDOR:ID<x>:IDEXt:HEXA	Sets the ID of each extended format of the OR condition of the CAN bus signal trigger in hexadecimal notation.	5-182
:TRIGger:ENHanced:CANBus:IDOR:ID<x>:IDEXt:PATtern	Sets the ID of each extended format of the OR condition of the CAN bus signal trigger in binary notation or queries the current setting.	5-182
:TRIGger:ENHanced:CANBus:IDOR:ID<x>:IDSTd?	Queries all settings related to the ID of each standard format of the OR condition of the CAN bus signal trigger.	5-182
:TRIGger:ENHanced:CANBus:IDOR:ID<x>:IDSTd:HEXA	Sets the ID of each standard format of the OR condition of the CAN bus signal trigger in hexadecimal notation.	5-182
:TRIGger:ENHanced:CANBus:IDOR:ID<x>:IDSTd:PATtern	Sets the ID of each standard format of the OR condition of the CAN bus signal trigger in binary notation or queries the current setting.	5-182
:TRIGger:ENHanced:CANBus:IDOR:ID<x>:MODE	Enables or disables each condition of the OR condition of the CAN bus signal trigger or queries the current setting.	5-183
:TRIGger:ENHanced:CANBus:IDOR:ID<x>:RTR	Sets each RTR of the OR condition of the CAN bus signal trigger or queries the current setting.	5-183
:TRIGger:ENHanced:CANBus:IDSTd?	Queries all settings related to the ID of the standard format of the CAN bus signal trigger.	5-183
:TRIGger:ENHanced:CANBus:IDSTd:HEXA	Sets the ID of the standard format of the CAN bus signal trigger in hexadecimal notation.	5-183
:TRIGger:ENHanced:CANBus:IDSTd:PATtern	Sets the ID of the standard format of the CAN bus signal trigger in binary notation or queries the current setting.	5-183
:TRIGger:ENHanced:CANBus:MODE	Sets the CAN bus signal trigger mode or queries the current setting.	5-183
:TRIGger:ENHanced:CANBus:RECEssive	Sets the recessive level (bus level) of the CAN bus signal trigger or queries the current setting.	5-183
:TRIGger:ENHanced:CANBus:RTR	Sets the RTR of the CAN bus signal trigger or queries the current setting.	5-183
:TRIGger:ENHanced:CANBus:SOURce	Sets the trigger source of the CAN bus signal trigger or queries the current setting.	5-184
:TRIGger:ENHanced:CANBus:SPOint	Sets the sample point of the CAN bus signal trigger or queries the current setting.	5-184
:TRIGger:ENHanced:I2Cbus?	Queries all settings related to the I <sup>2</sup> C bus trigger.	5-184
:TRIGger:ENHanced:I2Cbus:ADATa?	Queries all settings related to the address of the I <sup>2</sup> C bus trigger.	5-184
:TRIGger:ENHanced:I2Cbus:ADATa:BIT10address?	Queries all settings related to the 10-bit address of the I <sup>2</sup> C bus trigger.	5-184
:TRIGger:ENHanced:I2Cbus:ADATa:BIT10address:HEXA	Sets the 10-bit address of the I <sup>2</sup> C bus trigger in hexadecimal notation.	5-184
:TRIGger:ENHanced:I2Cbus:ADATa:BIT10address:PATtern	Sets the 10-bit address of the I <sup>2</sup> C bus trigger in binary notation or queries the current setting.	5-184
:TRIGger:ENHanced:I2Cbus:ADATa:BIT7Address?	Queries all settings related to the 7-bit address of the I <sup>2</sup> C bus trigger.	5-185

## 5.1 List of Commands

Command	Function	Page
:TRIGger:ENHanced:I2CBus:ADATa:BIT7Address:HEXA	Sets the 7-bit address of the I <sup>2</sup> C bus trigger in hexadecimal notation.	5-185
:TRIGger:ENHanced:I2CBus:ADATa:BIT7Address:PATtern	Sets the 7-bit address of the I <sup>2</sup> C bus trigger in binary notation or queries the current setting.	5-185
:TRIGger:ENHanced:I2CBus:ADATa:BIT7APsub?	Queries all settings related to the 7-bit + Sub address of the I <sup>2</sup> C bus trigger.	5-185
:TRIGger:ENHanced:I2CBus:ADATa:BIT7APsub:ADDRESS?	Queries all settings related to the 7-bit address of the 7-bit + Sub address of the I <sup>2</sup> C bus trigger.	5-185
:TRIGger:ENHanced:I2CBus:ADATa:BIT7APsub:ADDRESS:HEXA	Sets the 7-bit address of the 7-bit + Sub address of the I <sup>2</sup> C bus trigger in hexadecimal notation.	5-185
:TRIGger:ENHanced:I2CBus:ADATa:BIT7APsub:ADDRESS:PATtern	Sets the 7-bit address of the 7-bit + Sub address of the I <sup>2</sup> C bus trigger in binary notation or queries the current setting.	5-185
:TRIGger:ENHanced:I2CBus:ADATa:BIT7APsub:SADDRESS?	Queries all settings related to the Sub address of the 7-bit + Sub address of the I <sup>2</sup> C bus trigger.	5-185
:TRIGger:ENHanced:I2CBus:ADATa:BIT7APsub:SADDRESS:HEXA	Sets the Sub address of the 7-bit + Sub address of the I <sup>2</sup> C bus trigger in hexadecimal notation.	5-186
:TRIGger:ENHanced:I2CBus:ADATa:BIT7APsub:SADDRESS:PATtern	Sets the Sub address of the 7-bit + Sub address of the I <sup>2</sup> C bus trigger in binary notation or queries the current setting.	5-186
:TRIGger:ENHanced:I2CBus:ADATa:TYPE	Sets the address type of the I <sup>2</sup> C bus trigger or queries the current setting.	5-186
:TRIGger:ENHanced:I2CBus:CLOCK?	Queries all settings related to the clock of the I <sup>2</sup> C bus trigger.	5-186
:TRIGger:ENHanced:I2CBus:CLOCK:SOURce	Sets the clock trace of the I <sup>2</sup> C bus trigger or queries the current setting.	5-186
:TRIGger:ENHanced:I2CBus:DATA?	Queries all settings related to the data of the I <sup>2</sup> C bus trigger.	5-186
:TRIGger:ENHanced:I2CBus:DATA:BYTE	Sets the number of data bytes of the I <sup>2</sup> C bus trigger or queries the current setting.	5-186
:TRIGger:ENHanced:I2CBus:DATA:CONDition	Sets the determination method (match or not match) of the data of the I <sup>2</sup> C bus trigger or queries the current setting.	5-186
:TRIGger:ENHanced:I2CBus:DATA:DPOSITion	Sets the position for comparing the data pattern of the I <sup>2</sup> C bus trigger or queries the current setting.	5-187
:TRIGger:ENHanced:I2CBus:DATA:HEXA<x>	Sets the data of the I <sup>2</sup> C bus trigger in hexadecimal notation.	5-187
:TRIGger:ENHanced:I2CBus:DATA:MODE	Enables/Disables the data conditions of the I <sup>2</sup> C bus trigger or queries the current setting.	5-187
:TRIGger:ENHanced:I2CBus:DATA:PATtern<x>	Sets the data of the I <sup>2</sup> C bus trigger in binary notation or queries the current setting.	5-187
:TRIGger:ENHanced:I2CBus:DATA:PMODE	Sets the pattern comparison start position mode of the data of the I <sup>2</sup> C bus trigger or queries the current setting.	5-187
:TRIGger:ENHanced:I2CBus:DATA:SOURce	Sets the data trace of the I <sup>2</sup> C bus trigger or queries the current setting.	5-187
:TRIGger:ENHanced:I2CBus:GCALl?	Queries all settings related to the general call of the I <sup>2</sup> C bus trigger.	5-187
:TRIGger:ENHanced:I2CBus:GCALl:BIT7maddress?	Queries all settings related to the 7-bit master address of the general call of the I <sup>2</sup> C bus trigger.	5-187
:TRIGger:ENHanced:I2CBus:GCALl:BIT7maddress:HEXA	Sets the 7-bit master address of the general call of the I <sup>2</sup> C bus trigger in hexadecimal notation.	5-188
:TRIGger:ENHanced:I2CBus:GCALl:BIT7maddress:PATtern	Sets the 7-bit master address of the general call of the I <sup>2</sup> C bus trigger in binary notation or queries the current setting.	5-188
:TRIGger:ENHanced:I2CBus:GCALl:SBYTE (Second Byte)	Sets the second byte type of the general call of the I <sup>2</sup> C bus trigger or queries the current setting.	5-188
:TRIGger:ENHanced:I2CBus:MODE	Sets the trigger mode of the I <sup>2</sup> C bus trigger or queries the current setting.	5-188
:TRIGger:ENHanced:I2CBus:NAIGNore?	Queries all settings related to the NON ACK ignore mode of the I <sup>2</sup> C bus trigger.	5-188
:TRIGger:ENHanced:I2CBus:NAIGNore:HSMODE	Sets whether to ignore NON ACK in high speed mode of the I <sup>2</sup> C bus trigger or queries the current setting.	5-188
:TRIGger:ENHanced:I2CBus:NAIGNore:RACCESS	Sets whether to ignore NON ACK in read access mode of the I <sup>2</sup> C bus trigger or queries the current setting.	5-188
:TRIGger:ENHanced:I2CBus:NAIGNore:SBYTE (Start Byte)	Sets whether to ignore NON ACK in the start byte of the I <sup>2</sup> C bus trigger or queries the current setting.	5-188
:TRIGger:ENHanced:I2CBus:SBHSMODE?	Queries all settings related to the start byte and high speed mode of the I <sup>2</sup> C bus trigger.	5-189
:TRIGger:ENHanced:I2CBus:SBHSMODE:TYPE	Sets the type of the start byte or high speed mode of the I <sup>2</sup> C bus trigger or queries the current setting.	5-189
:TRIGger:ENHanced:LINBus?	Queries all settings related to the LIN bus trigger or queries the current setting.	5-189

## 5.1 List of Commands

Command	Function	Page
:TRIGger:ENHanced:LINBus:BRATe	Sets the LIN bus signal trigger bitrate (data transfer rate) or queries the current setting.	5-189
:TRIGger:ENHanced:LINBus:SOURce	Sets the LIN bus signal trigger source or queries the current setting.	5-189
:TRIGger:ENHanced:SPATtern? (Serial Pattern)	Queries all settings related to the serial pattern trigger.	5-189
:TRIGger:ENHanced:SPATtern: BITRate	Sets the bit rate of the serial pattern trigger or queries the current setting.	5-189
:TRIGger:ENHanced:SPATtern: CLEAr	Clears the entire pattern of the serial pattern trigger (to don't care).	5-189
:TRIGger:ENHanced:SPATtern: CLOCK?	Queries all settings related to clock of the serial pattern trigger.	5-189
:TRIGger:ENHanced:SPATtern: CLOCK:MODE	Enables/Disables the clock of the serial pattern trigger or queries the current setting.	5-189
:TRIGger:ENHanced:SPATtern: CLOCK:POLarity	Sets the polarity of the clock trace of the serial pattern trigger or queries the current setting.	5-190
:TRIGger:ENHanced:SPATtern: CLOCK:SOURce	Sets the clock trace of the serial pattern trigger or queries the current setting.	5-190
:TRIGger:ENHanced:SPATtern:CS	Enables/Disables the chip select of the serial pattern trigger or queries the current setting.	5-190
:TRIGger:ENHanced:SPATtern: DATA?	Queries all settings related to data of the serial pattern trigger.	5-190
:TRIGger:ENHanced:SPATtern: DATA:ACTive	Sets the active level of the data of the serial pattern trigger or queries the current setting.	5-190
:TRIGger:ENHanced:SPATtern: DATA:SOURce	Sets the data trace of the serial pattern trigger or queries the current setting.	5-190
:TRIGger:ENHanced:SPATtern:HEXA	Sets the pattern of the serial pattern trigger in hexadecimal notation.	5-190
:TRIGger:ENHanced:SPATtern: LATCh?	Queries all settings related to latch of the serial pattern trigger.	5-190
:TRIGger:ENHanced:SPATtern: LATCh:POLarity	Sets the polarity of the latch trace of the serial pattern trigger or queries the current setting.	5-191
:TRIGger:ENHanced:SPATtern: LATCh:SOURce	Sets the latch trace of the serial pattern trigger or queries the current setting.	5-191
:TRIGger:ENHanced:SPATtern: PATtern	Sets the pattern of the serial pattern trigger in binary notation or queries the current setting.	5-191
:TRIGger:ENHanced:SPIBus?	Queries all settings related to the SPI bus trigger.	5-191
:TRIGger:ENHanced:SPIBus: BITorder	Sets the bit order of the SPI bus trigger or queries the current setting.	5-191
:TRIGger:ENHanced:SPIBus:CLOCK?	Queries all settings related to the clock of the SPI bus trigger.	5-191
:TRIGger:ENHanced:SPIBus:CLOCK: POLarity	Sets the polarity of the clock trace of the SPI bus trigger or queries the current setting.	5-191
:TRIGger:ENHanced:SPIBus:CLOCK: SOURce	Sets the clock trace of the SPI bus trigger or queries the current setting.	5-192
:TRIGger:ENHanced:SPIBus:CS?	Queries all settings related to the chip select of the SPI bus trigger.	5-192
:TRIGger:ENHanced:SPIBus:CS: ACTive	Sets the active level of the chip select of the SPI bus trigger or queries the current setting.	5-192
:TRIGger:ENHanced:SPIBus:CS: SOURce	Sets the chip select trace of the SPI bus trigger or queries the current setting.	5-192
:TRIGger:ENHanced:SPIBus: DATA<x>?	Queries all settings related to the data of the SPI bus trigger.	5-192
:TRIGger:ENHanced:SPIBus: DATA<x>:BYTE	Sets the number of bytes of the data of the SPI bus trigger or queries the current setting.	5-192
:TRIGger:ENHanced:SPIBus: DATA<x>:CONDition	Sets the determination method (match or not match) of the data of the SPI bus trigger or queries the current setting.	5-192
:TRIGger:ENHanced:SPIBus: DATA<x>:DPOSITion	Sets the pattern comparison start position of the data of the SPI bus trigger or queries the current setting.	5-192
:TRIGger:ENHanced:SPIBus: DATA<x>:HEXA<x>	Sets the data of the SPI bus trigger in hexadecimal notation.	5-193
:TRIGger:ENHanced:SPIBus: DATA<x>:PATtern<x>	Sets the data of the SPI bus trigger in binary notation or queries the current setting.	5-193
:TRIGger:ENHanced:SPIBus: DATA<x>:SOURce	Sets the trace of the data of the SPI bus trigger or queries the current setting.	5-193
:TRIGger:ENHanced:SPIBus:MODE	Sets the wiring system of the SPI bus trigger (three-wire or four-wire) or queries the current setting.	5-193
:TRIGger:ENHanced:TV?	Queries all settings related to the TV trigger.	5-193



## 5.1 List of Commands

Command	Function	Page
:TRIGger:ENHanced:TV:COUPling?	Queries the trigger coupling of the TV trigger.	5-193
:TRIGger:ENHanced:TV:CUSTomize	Turns ON/OFF the sync guard function of the TV trigger or queries the current setting.	5-193
:TRIGger:ENHanced:TV:FIELD	Sets the field of the TV trigger or queries the current setting.	5-193
:TRIGger:ENHanced:TV:FRAMe	Sets the frame skip function of the TV trigger or queries the current setting.	5-193
:TRIGger:ENHanced:TV:{HDTV NTSC PAL SDTV USERdefine}?	Queries all settings related to the TV trigger mode.	5-194
:TRIGger:ENHanced:TV:{HDTV NTSC PAL SDTV}:HFRejection?(HighFrequencyREJECTION)	Queries the low pass filter (HF rejection) of the TV trigger.	5-194
:TRIGger:ENHanced:TV:{HDTV NTSC PAL SDTV USERdefine}:LINE	Sets the line for activating the TV trigger or queries the current setting.	5-194
:TRIGger:ENHanced:TV:{HDTV NTSC PAL SDTV USERdefine}:POLarity	Sets the input polarity of the TV trigger or queries the current setting.	5-194
:TRIGger:ENHanced:TV:LEVel	Sets the trigger level of the TV trigger or queries the current setting.	5-194
:TRIGger:ENHanced:TV:SGUard	Sets the sync guard of the TV trigger or queries the current setting.	5-194
:TRIGger:ENHanced:TV:SOURce	Sets the trigger source of the TV trigger or queries the current setting.	5-194
:TRIGger:ENHanced:TV:TYPE	Sets the input type of the TV trigger or queries the current setting.	5-194
:TRIGger:ENHanced:TV:USERdefine:DEFinition	Sets the user-defined resolution or queries the current setting.	5-195
:TRIGger:ENHanced:TV:USERdefine:HFRejection(HighFrequencyREJECTION)	Sets the user-defined low pass filter (HF rejection) or queries the current setting.	5-195
:TRIGger:ENHanced:TV:USERdefine:HSYNc(Hsync Freq)	Sets the user-defined horizontal sync signal or queries the current setting.	5-195
:TRIGger:ENHanced:UART?	Queries all settings related to the UART signal trigger.	5-195
:TRIGger:ENHanced:UART:BRATe	Sets the UART signal trigger bit rate (data transfer rate) or queries the current setting.	5-195
:TRIGger:ENHanced:UART:FORMat	Sets the UART signal trigger format or queries the current setting.	5-195
:TRIGger:ENHanced:UART:POLarity	Sets the UART signal trigger polarity or queries the current setting.	5-195
:TRIGger:ENHanced:UART:SOURce	Sets the UART signal trigger source or queries the current setting.	5-195
:TRIGger:ENHanced:UART:SPOint	Sets the UART signal trigger sample point or queries the current setting.	5-195
:TRIGger:ESTate?	Queries all settings related to the edge/state trigger.	5-196
:TRIGger:ESTate:EOR?	Queries all settings related to the OR trigger.	5-196
:TRIGger:ESTate:EOR:CHANnel<x>	Sets the channel polarity of the OR trigger or queries the current setting.	5-196
:TRIGger:ESTate:POLarity	Sets the polarity of the edge/state trigger or queries the current setting.	5-196
:TRIGger:ESTate:SOURce	Sets the trigger source of the edge/state trigger or queries the current setting.	5-196
:TRIGger:HOLDoff	Sets the hold off time or queries the current setting.	5-196
:TRIGger:MODE	Sets the trigger mode or queries the current setting.	5-196
:TRIGger:POSition	Sets the trigger position or queries the current setting.	5-196
:TRIGger:SCOUNT (Single(N) Count)	Sets the number of times the trigger is to be activated when the trigger mode is Single(N) or queries the current setting.	5-196
:TRIGger:SOURce?	Queries all settings related to the trigger source.	5-197
:TRIGger:SOURce:CHANnel<x>?	Queries all settings related to the channel of the trigger source.	5-197
:TRIGger:SOURce:CHANnel<x>:COUPling	Sets the trigger coupling of the channel or queries the current setting.	5-197
:TRIGger:SOURce:CHANnel<x>:HFRejection(HighFrequencyREJECTION)	Sets the low pass filter (HF rejection) of the channel or queries the current setting.	5-197
:TRIGger:SOURce:CHANnel<x>:HYSTeresis	Sets the hysteresis of the channel or queries the current setting.	5-197
:TRIGger:SOURce:CHANnel<x>:LEVel	Sets the trigger level of the channel or queries the current setting.	5-197
:TRIGger:SOURce:CHANnel<x>:STATe	Sets the condition to be satisfied of the channel or queries the current setting.	5-198
:TRIGger:SOURce:CHANnel<x>:WIDTh	Sets the window trigger width of the channel or queries the current setting.	5-198
:TRIGger:SOURce:CHANnel<x>:WINDow	Turns ON/OFF the window of the channel or queries the current setting.	5-198
:TRIGger:SOURce:EXTernal?	Queries all settings related to the external trigger.	5-198
:TRIGger:SOURce:EXTernal:LEVel	Sets the trigger level of the external trigger or queries the current setting.	5-198

## 5.1 List of Commands

Command	Function	Page
:TRIGger:SOURce:EXTernal:PROBe	Sets the probe attenuation of the external trigger or queries the current setting.	5-198
:TRIGger:SOURce:LOGic	Sets the trigger source logic or queries the current setting.	5-198
:TRIGger:TYPE	Sets the trigger type or queries the current setting.	5-198
:TRIGger:WIDTH?	Queries all settings related to the pulse width trigger.	5-198
:TRIGger:WIDTH:MODE	Sets the determination mode of the pulse width trigger or queries the current setting.	5-199
:TRIGger:WIDTH:POLarity	Sets the polarity of the pulse width trigger or queries the current setting.	5-199
:TRIGger:WIDTH:SOURce	Sets the trigger source of the pulse width trigger or queries the current setting.	5-199
:TRIGger:WIDTH:TIME<x>	Sets the pulse width of the pulse width trigger or queries the current setting.	5-199

### WAVeform Group

:WAVeform?	Queries all information about the waveform data.	5-200
:WAVeform:BITS?	Queries the bit length of the waveform data specified by ":WAVeform:TRACe".	5-200
:WAVeform:BYTeorder	Sets the transmission order when using word format of two bytes or more or queries the current setting.	5-200
:WAVeform:END	Sets the last data point of the waveform specified by :WAVeform:TRACe or queries the current setting.	5-200
:WAVeform:FORMat	Sets the format of the data to be transmitted or queries the current setting.	5-200
:WAVeform:LENGth?	Queries the total number of points of the waveform specified by ":WAVeform:TRACe".	5-200
:WAVeform:OFFSet?	Queries the offset value when converting the waveform data specified by :WAVeform:TRACe to physical values.	5-200
:WAVeform:POSition?	Queries the vertical axis position used for converting to voltage when RBYTe is specified with :WAVeform:FORMat.	5-200
:WAVeform:RANGe?	Queries the range value when converting the waveform data specified by :WAVeform:TRACe to physical values.	5-200
:WAVeform:RECORD	Sets the target record number for the commands in the WAVeform group or queries the current setting.	5-200
:WAVeform:RECORD? MINimum	Queries the minimum record number of the history of the target channel.	5-201
:WAVeform:SEND?	Queries the waveform data specified by ":WAVeform:TRACe".	5-201
:WAVeform:SIGN?	Queries the existence of a sign when querying the waveform data specified by :WAVeform:TRACe using binary data.	5-201
:WAVeform:SRATE? (Sample RATE)	Queries the sample rate of the record specified by :WAVeform:RECORD.	5-201
:WAVeform:START	Sets the first data point of the waveform specified by :WAVeform:TRACe or queries the current setting.	5-201
:WAVeform:TRACe	Sets the target waveform or queries the current setting.	5-201
:WAVeform:TRIGger?	Queries the trigger position of the record specified by :WAVeform:RECORD.	5-201
:WAVeform:TYPE?	Queries the acquisition mode of the waveform specified by :WAVeform:TRACe.	5-201

### ZOOM Group

:ZOOM?	Queries all settings related to the waveform zoom.	5-202
:ZOOM:ALLocation<x>?	Queries all settings related to the zoom source waveform.	5-202
:ZOOM:ALLocation<x>:ALLon	Sets all waveforms to be zoomed.	5-202
:ZOOM:ALLocation<x>:TRACe<x>	Turns ON/OFF the trace you wish to zoom or queries the current setting.	5-202
:ZOOM:FORMat<x>	Sets the display format of the zoom waveform or queries the current setting.	5-202
:ZOOM:HLINKage	Turns ON/OFF the horizontal link or queries the current setting.	5-202
:ZOOM:HORizontal<x>?	Queries all settings related to the horizontal zoom.	5-202
:ZOOM:HORizontal<x>:ASCROLL?	Queries all settings related to the auto scroll function.	5-202
:ZOOM:HORizontal<x>:ASCROLL: JUMP	Moves the zoom center position to the left or right edge of the main screen.	5-202
:ZOOM:HORizontal<x>:ASCROLL: SPEEd	Sets the auto scroll speed or queries the current setting.	5-202
:ZOOM:HORizontal<x>:ASCROLL: START	Starts auto scrolling.	5-203
:ZOOM:HORizontal<x>:ASCROLL: STOP	Stops auto scrolling.	5-203
:ZOOM:HORizontal<x>:MAG	Sets the horizontal zoom magnification or queries the current setting.	5-203
:ZOOM:HORizontal<x>:POSition	Sets the horizontal zoom center position or queries the current setting.	5-203
:ZOOM:MODE	Sets the zoom waveform display format or queries the current setting.	5-203

## 5.1 List of Commands

Command	Function	Page
:ZOOM:TYPE<x>	Sets the zoom type or queries the current setting.	5-203
:ZOOM:VERTical<x>?	Queries all settings related to the vertical zoom.	5-203
:ZOOM:VERTical<x>:INITialize	Initializes the vertical zoom.	5-203
:ZOOM:VERTical<x>:MAG	Sets the vertical zoom magnification or queries the current setting.	5-203
:ZOOM:VERTical<x>:POSition	Sets the vertical zoom position or queries the current setting.	5-203
:ZOOM:VERTical<x>:TRACe	Sets the trace you wish to display on the vertical zoom screen or queries the current setting.	5-203
:ZOOM:VLINKage	Turns ON/OFF the vertical link or queries the current setting.	5-203

### Common Command Group

*CAL? (CALibrate)	Performs calibration and queries the result.	5-204
*CLS (CLear Status)	Clears the standard event register, extended event register, and error queue.	5-204
*ESE (standard Event Status Enable register)	Sets the standard event enable register or queries the current setting.	5-204
*ESR? (standard Event Status Register)	Queries the standard event register and clears the register.	5-204
*IDN? (IDeNtify)	Queries the instrument model.	5-204
*LRN? (LeaRN)	Queries collectively the current settings of the following command groups.	5-205
*OPC (OPeration Complete)	Sets bit 0 (OPC bit) of the standard event register to 1 upon the completion of the specified overlap command.	5-206
*OPC? (OPeration Complete)	If *OPC? is transmitted and the specified overlap command is completed, ASCII code 1 is returned.	5-206
*OPT? (OPTion)	Queries the installed options.	5-206
*PSC (Power-on Status Clear)	Sets whether or not to clear the registers below at power on or queries the current setting. The register is cleared when the value rounded to an integer is a non-zero value.	5-206
*RST (ReSeT)	Initializes the settings.	5-206
*SRE (Service Request Enable register)	Sets the service request enable register or queries the current setting.	5-206
*STB? (STatus Byte)	Queries the status byte register.	5-206
*TST?	Performs a self-test and queries the result. The self test involves internal memory tests.	5-207
*WAI (WAIt)	Holds the subsequent command until the completion of the specified overlap operation.	5-207

## 5.2 ACQUIRE Group

### **:ACQUIRE?**

Function Queries all settings related to the waveform acquisition.

Syntax :ACQUIRE?

Example :ACQUIRE? -> :ACQUIRE:AVERAGE:COUNT 2;  
EWEIGHT 16;:ACQUIRE:HRMODE 0;  
INTERLEAVE 0;INTERPOLATE 1;MODE NORMAL;  
REPETITIVE 0;RLENGTH 12500

### **:ACQUIRE:AVERAGE?**

Function Queries all settings related to averaging and the waveform acquisition count.

Syntax :ACQUIRE:AVERAGE?

Example :ACQUIRE:AVERAGE? -> :ACQUIRE:AVERAGE:  
COUNT 2;EWEIGHT 16

### **:ACQUIRE:AVERAGE:COUNT**

Function Sets the waveform acquisition count of averaging mode or queries the current setting.

Syntax :ACQUIRE:AVERAGE:COUNT {<NRf>}  
:ACQUIRE:AVERAGE:COUNT?  
<NRf> = 2 to 65536 in 2<sup>n</sup> steps

Example :ACQUIRE:AVERAGE:COUNT 2  
:ACQUIRE:AVERAGE:COUNT? -> :ACQUIRE:  
AVERAGE:COUNT 2

### **:ACQUIRE:AVERAGE:EWEIGHT**

Function Sets the attenuation constant when averaging mode is used infinitely or queries the current setting.

Syntax :ACQUIRE:AVERAGE:EWEIGHT {<NRf>}  
:ACQUIRE:AVERAGE:EWEIGHT?  
<NRf> = 2 to 1024 in 2<sup>n</sup> steps

Example :ACQUIRE:AVERAGE:EWEIGHT 16  
:ACQUIRE:AVERAGE:EWEIGHT? -> :ACQUIRE:  
AVERAGE:EWEIGHT 16

### **:ACQUIRE:HRMODE**

Function Turns ON/OFF the high resolution mode or queries the current setting.

Syntax :ACQUIRE:HRMODE {<Boolean>}  
:ACQUIRE:HRMODE?

Example :ACQUIRE:HRMODE ON  
:ACQUIRE:HRMODE? -> :ACQUIRE:HRMODE 1

### **:ACQUIRE:INTERLEAVE**

Function Turns ON/OFF interleave or queries the current setting.

Syntax :ACQUIRE:INTERLEAVE {<Boolean>}  
:ACQUIRE:INTERLEAVE?

Example :ACQUIRE:INTERLEAVE ON  
:ACQUIRE:INTERLEAVE? -> :ACQUIRE:  
INTERLEAVE 1

### **:ACQUIRE:INTERPOLATE**

Function Turns ON/OFF data interpolation or queries the current setting.

Syntax :ACQUIRE:INTERPOLATE {<Boolean>}  
:ACQUIRE:INTERPOLATE?

Example :ACQUIRE:INTERPOLATE ON  
:ACQUIRE:INTERPOLATE? -> :ACQUIRE:  
INTERPOLATE 1

### **:ACQUIRE:MODE**

Function Sets the waveform acquisition mode or queries the current setting.

Syntax :ACQUIRE:MODE {AVERAGE|ENVELOPE|NORMAL}  
:ACQUIRE:MODE?

Example :ACQUIRE:MODE NORMAL  
:ACQUIRE:MODE? -> :ACQUIRE:MODE NORMAL

### **:ACQUIRE:REPETITIVE**

Function Turns ON/OFF the repetitive sampling or queries the current setting.

Syntax :ACQUIRE:REPETITIVE {<Boolean>}  
:ACQUIRE:REPETITIVE?

Example :ACQUIRE:REPETITIVE ON  
:ACQUIRE:REPETITIVE? -> :ACQUIRE:  
REPETITIVE 1

### **:ACQUIRE:RLENGTH**

Function Sets the record length or queries the current setting.

Syntax :ACQUIRE:RLENGTH {<NRf>}  
:ACQUIRE:RLENGTH?

<NRf> = See the DL9000 User's Manual.  
Example :ACQUIRE:RLENGTH 6250000  
:ACQUIRE:RLENGTH? -> :ACQUIRE:  
RLENGTH 6250000

## 5.3 ANALysis Group

### :ANALysis?

**Function** Queries all settings related to the analysis function.

**Syntax** :ANALysis?Example  
 :ANALYSIS? -> :ANALYSIS:  
 AHISTOGRAM1:HORIZONTAL 0.000E+00,  
 0.000E+00;MEASURE:CURSOR:BASIC:C1:  
 STATE 1;;ANALYSIS:AHISTOGRAM1:MEASURE:  
 CURSOR:BASIC:C2:STATE 1;;ANALYSIS:  
 AHISTOGRAM1:MEASURE:CURSOR:BASIC:DC:  
 STATE 1;;ANALYSIS:AHISTOGRAM1:MEASURE:  
 CURSOR:CALCULATION:DEFINE1 "C1";  
 DEFINE2 "C2";DEFINE3 "C1";DEFINE4 "C2";  
 STATE1 1;STATE2 1;STATE3 1;STATE4 1;;  
 ANALYSIS:AHISTOGRAM1:MEASURE:CURSOR:  
 HLINKAGE 1;HPOSITION1 0.000E+00;  
 HPOSITION2 1.000E+00;VLINKAGE 1;  
 VPOSITION1 1.000E+00;  
 VPOSITION2 1.000E+00;;ANALYSIS:  
 AHISTOGRAM1:MEASURE:MODE CURSOR;  
 PARAMETER:AREA1:MAXIMUM:STATE 1;;  
 ANALYSIS:AHISTOGRAM1:MEASURE:PARAMETER:  
 AREA1:MEAN:STATE 1;;ANALYSIS:  
 AHISTOGRAM1:MEASURE:PARAMETER:AREA1:  
 MEDIAN:STATE 1;;ANALYSIS:AHISTOGRAM1:  
 MEASURE:PARAMETER:AREA1:MINIMUM:STATE  
 1;;ANALYSIS:AHISTOGRAM1:MEASURE:  
 PARAMETER:AREA1:PEAK:STATE 1;;ANALYSIS:  
 AHISTOGRAM1:MEASURE:PARAMETER:AREA1:  
 RMS:STATE 1;;ANALYSIS:AHISTOGRAM1:  
 MEASURE:PARAMETER:AREA1:SD2INTEG:  
 STATE 1;;ANALYSIS:AHISTOGRAM1:MEASURE:  
 PARAMETER:AREA1:SD3INTEG:STATE 1;;  
 ANALYSIS:AHISTOGRAM1:MEASURE:PARAMETER:  
 AREA1:SDEVIATION:STATE 1;;ANALYSIS:  
 AHISTOGRAM1:MEASURE:PARAMETER:AREA1:  
 SDINTEG:STATE 1.....

### :ANALysis:AHISTogram<x>?

**Function** Queries all settings related to the accumulated histogram function.

**Syntax** :ANALysis:AHISTogram<x>?  
 <x> = 1 or 2

**Example** :ANALYSIS:AHISTOGRAM1? -> :ANALYSIS:  
 AHISTOGRAM1:HORIZONTAL 0.000E+00,  
 0.000E+00;MEASURE:CURSOR:BASIC:C1:  
 STATE 1;;ANALYSIS:AHISTOGRAM1:  
 MEASURE:CURSOR:BASIC:C2:STATE 1;;  
 ANALYSIS:AHISTOGRAM1:MEASURE:CURSOR:  
 BASIC:DC:STATE 1;;ANALYSIS:AHISTOGRAM1:  
 MEASURE:CURSOR:CALCULATION:  
 DEFINE1 "C1";DEFINE2 "C2";DEFINE3 "C1";  
 DEFINE4 "C2";STATE1 1;STATE2 1;  
 STATE3 1;STATE4 1;;ANALYSIS:  
 AHISTOGRAM1:MEASURE:CURSOR:HLINKAGE 1;  
 HPOSITION1 0.000E+00;  
 HPOSITION2 1.000E+00;VLINKAGE 1;  
 VPOSITION1 1.000E+00;  
 VPOSITION2 1.000E+00;;ANALYSIS:  
 AHISTOGRAM1:MEASURE:MODE CURSOR;  
 PARAMETER:AREA1:MAXIMUM:STATE 1;;  
 ANALYSIS:AHISTOGRAM1:MEASURE:PARAMETER:  
 AREA1:MEAN:STATE 1;;ANALYSIS:  
 AHISTOGRAM1:MEASURE:PARAMETER:AREA1:  
 MEDIAN:STATE 1;;ANALYSIS:AHISTOGRAM1:  
 MEASURE:PARAMETER:AREA1:MINIMUM:  
 STATE 1;;ANALYSIS:AHISTOGRAM1:MEASURE:  
 PARAMETER:AREA1:PEAK:STATE 1;;ANALYSIS:  
 AHISTOGRAM1:MEASURE:PARAMETER:AREA1:  
 RMS:STATE 1;;ANALYSIS:AHISTOGRAM1:  
 MEASURE:PARAMETER:AREA1:SD2INTEG:  
 STATE 1;;ANALYSIS:AHISTOGRAM1:MEASURE:  
 PARAMETER:AREA1:SD3INTEG:STATE 1;;A  
 ANALYSIS:AHISTOGRAM1:MEASURE:PARAMETER:  
 AREA1:SDEVIATION:STATE 1;;ANALYSIS:  
 AHISTOGRAM1:MEASURE:PARAMETER:AREA1:  
 SDINTEG:STATE 1.....

### :ANALysis:AHISTogram<x>:HORizontal

**Function** Sets the horizontal range of the accumulated histogram or queries the current setting.

**Syntax** :ANALysis:AHISTogram<x>:HORizontal  
 {<Nrf>,<Nrf>}  
 :ANALysis:AHISTogram<x>:HORizontal?  
 <x> = 1 or 2  
 <Nrf> = -4 to 4 (div)

**Example** :ANALYSIS:AHISTOGRAM1:HORIZONTAL 0,1  
 :ANALYSIS:AHISTOGRAM1:HORIZONTAL?  
 -> :ANALYSIS:AHISTOGRAM1:  
 HORIZONTAL 0.000E+00,0.000E+00

### 5.3 ANALysis Group

#### **:ANALysis:AHISTogram<x>:MEASure?**

**Function** Queries all settings related automated measurement of the accumulated histogram.

**Syntax** :ANALysis:AHISTogram<x>:MEASure?  
<x> = 1 or 2

**Example** :ANALYSIS:AHISTOGRAM1:MEASURE? -> :  
ANALYSIS:AHISTOGRAM1:MEASURE:CURSOR:  
BASIC:C1:STATE 1;:ANALYSIS:AHISTOGRAM1:  
MEASURE:CURSOR:BASIC:C2:STATE 1;:  
ANALYSIS:AHISTOGRAM1:MEASURE:CURSOR:  
BASIC:DC:STATE 1;:ANALYSIS:AHISTOGRAM1:  
MEASURE:CURSOR:CALCULATION:  
DEFINE1 "C1";DEFINE2 "C2";DEFINE3 "C1";  
DEFINE4 "C2";STATE1 1;STATE2 1;  
STATE3 1;STATE4 1;:ANALYSIS:  
AHISTOGRAM1:MEASURE:CURSOR:HLINKAGE 1;  
HPOSITION1 0.000E+00;  
HPOSITION2 1.000E+00;VLINKAGE 1;  
VPOSITION1 0.000E+00;  
VPOSITION2 1.000E+00;:ANALYSIS:  
AHISTOGRAM1:MEASURE:MODE CURSOR;  
PARAMETER:AREA1:MAXIMUM:STATE 1;:  
ANALYSIS:AHISTOGRAM1:MEASURE:PARAMETER:  
AREA1:MEAN:STATE 1;:ANALYSIS:  
AHISTOGRAM1:MEASURE:PARAMETER:AREA1:  
MEDIAN:STATE 1;:ANALYSIS:AHISTOGRAM1:  
MEASURE:PARAMETER:AREA1:MINIMUM:  
STATE 1;:ANALYSIS:AHISTOGRAM1:MEASURE:  
PARAMETER:AREA1:PEAK:STATE 1;:ANALYSIS:  
AHISTOGRAM1:MEASURE:PARAMETER:AREA1:  
RMS:STATE 1;:ANALYSIS:AHISTOGRAM1:  
MEASURE:PARAMETER:AREA1:SD2INTEG:  
STATE 1;:ANALYSIS:AHISTOGRAM1:MEASURE:  
PARAMETER:AREA1:SD3INTEG:STATE 1;:  
ANALYSIS:AHISTOGRAM1:MEASURE:PARAMETER:  
AREA1:SDEVIATION:STATE 1;:ANALYSIS:  
AHISTOGRAM1:MEASURE:PARAMETER:AREA1:  
SDINTEG:STATE 1.....

#### **:ANALysis:AHISTogram<x>:MEASure: CURSor?**

**Function** Queries all settings related to cursor measurement of the accumulated histogram.

**Syntax** :ANALysis:AHISTogram<x>:MEASure:CURSor?  
<x> = 1 or 2

**Example** :ANALYSIS:AHISTOGRAM1:MEASURE:CURSOR?  
-> :ANALYSIS:AHISTOGRAM1:MEASURE:  
CURSOR:BASIC:C1:STATE 1;:ANALYSIS:  
AHISTOGRAM1:MEASURE:CURSOR:BASIC:C2:  
STATE 1;:ANALYSIS:AHISTOGRAM1:MEASURE:  
CURSOR:BASIC:DC:STATE 1;:ANALYSIS:  
AHISTOGRAM1:MEASURE:CURSOR:CALCULATION:  
DEFINE1 "C1";DEFINE2 "C2";DEFINE3 "C1";  
DEFINE4 "C2";STATE1 1;STATE2 1;  
STATE3 1;STATE4 1;:ANALYSIS:  
AHISTOGRAM1:MEASURE:CURSOR:HLINKAGE 1;  
HPOSITION1 0.000E+00;  
HPOSITION2 1.000E+00;VLINKAGE 1;  
VPOSITION1 0.000E+00;  
VPOSITION2 1.000E+00

#### **:ANALysis:AHISTogram<x>:MEASure: CURSor:BASic?**

**Function** Queries all settings related to basic items of the cursor of the accumulated histogram.

**Syntax** :ANALysis:AHISTogram<x>:MEASure:  
CURSor:BASic?  
<x> = 1 or 2

**Example** :ANALYSIS:AHISTOGRAM1:MEASURE:CURSOR:  
BASIC? -> :ANALYSIS:AHISTOGRAM1:  
MEASURE:CURSOR:BASIC:C1:STATE 1;:  
ANALYSIS:AHISTOGRAM1:MEASURE:CURSOR:  
BASIC:C2:STATE 1;:ANALYSIS:AHISTOGRAM1:  
MEASURE:CURSOR:BASIC:DC:STATE 1

#### **:ANALysis:AHISTogram<x>:MEASure: CURSor[:BASic]:ALL**

**Function** Turns ON/OFF all basic items of the cursor of the accumulated histogram.

**Syntax** :ANALysis:AHISTogram<x>:MEASure:  
CURSor[:BASic]:ALL {<Boolean>}  
<x> = 1 or 2

**Example** :ANALYSIS:AHISTOGRAM1:MEASURE:CURSOR:  
BASIC:ALL ON

**:ANALysis:AHISTogram<x>:MEASure:  
CURSor[:BASic]:C<x>?**

Function Queries all settings related to the cursor of the accumulated histogram.

Syntax :ANALysis:AHISTogram<x>:MEASure:  
CURSor[:BASic]:C<x>?  
<x> of AHISTogram<x> = 1 or 2  
<x> of C<x> = 1 or 2

Example :ANALYSIS:AHISTOGRAM1:MEASURE:CURSOR:  
BASIC:C1? -> :ANALYSIS:AHISTOGRAM1:  
MEASURE:CURSOR:BASIC:C1:STATE 1

**:ANALysis:AHISTogram<x>:MEASure:  
CURSor[:BASic]:C<x>:STATe**

Function Turns ON/OFF the cursor of the accumulated histogram or queries the current setting.

Syntax :ANALysis:AHISTogram<x>:MEASure:  
CURSor[:BASic]:C<x>:STATe {<Boolean>}  
:ANALysis:AHISTogram<x>:MEASure:  
CURSor[:BASic]:C<x>:STATe?  
<x> of AHISTogram<x> = 1 or 2  
<x> of C<x> = 1 or 2

Example :ANALYSIS:AHISTOGRAM1:MEASURE:CURSOR:  
BASIC:C1:STATE ON  
:ANALYSIS:AHISTOGRAM1:MEASURE:CURSOR:  
BASIC:C1:STATE? -> :ANALYSIS:  
AHISTOGRAM1:MEASURE:CURSOR:  
BASIC:C1:STATE 1

**:ANALysis:AHISTogram<x>:MEASure:  
CURSor[:BASic]:C<x>:VALue?**

Function Queries the measured value of the cursor of the accumulated histogram.

Syntax :ANALysis:AHISTogram<x>:MEASure:  
CURSor[:BASic]:C<x>:VALue?  
<x> of AHISTogram<x> = 1 or 2  
<x> of C<x> = 1 or 2

Example :ANALYSIS:AHISTOGRAM1:MEASURE:CURSOR:  
BASIC:C1:VALUE? -> :ANALYSIS:  
AHISTOGRAM1:MEASURE:CURSOR:  
BASIC:C1:VALUE 1.000E+00

**:ANALysis:AHISTogram<x>:MEASure:  
CURSor[:BASic]:DC?**

Function Queries all settings related to measured values between cursors of the accumulated histogram.

Syntax :ANALysis:AHISTogram<x>:MEASure:  
CURSor[:BASic]:DC?  
<x> = 1 or 2

Example :ANALYSIS:AHISTOGRAM1:MEASURE:CURSOR:  
BASIC:DC? -> :ANALYSIS:AHISTOGRAM1:  
MEASURE:CURSOR:BASIC:DC:STATE 1

**:ANALysis:AHISTogram<x>:MEASure:  
CURSor[:BASic]:DC:STATe**

Function Turns ON/OFF the measured values between cursors of the accumulated histogram or queries the current setting.

Syntax :ANALysis:AHISTogram<x>:MEASure:  
CURSor[:BASic]:DC:STATe {<Boolean>}  
:ANALysis:AHISTogram<x>:MEASure:  
CURSor[:BASic]:DC:STATe?  
<x> = 1 or 2

Example :ANALYSIS:AHISTOGRAM1:MEASURE:CURSOR:  
BASIC:DC:STATE ON  
:ANALYSIS:AHISTOGRAM1:MEASURE:CURSOR:  
BASIC:DC:STATE? -> :ANALYSIS:  
AHISTOGRAM1:  
MEASURE:CURSOR:BASIC:DC:STATE 1

**:ANALysis:AHISTogram<x>:MEASure:  
CURSor[:BASic]:DC:VALue?**

Function Queries the measured value between cursors of the accumulated histogram.

Syntax :ANALysis:AHISTogram<x>:MEASure:  
CURSor[:BASic]:DC:VALue?  
<x> = 1 or 2

Example :ANALYSIS:AHISTOGRAM1:MEASURE:CURSOR:  
BASIC:DC:VALUE? -> :ANALYSIS:  
AHISTOGRAM1:MEASURE:CURSOR:  
BASIC:DC:VALUE 1.000E+00

**:ANALysis:AHISTogram<x>:MEASure:  
CURSor:CALCulation?**

Function Queries all settings related to calculation items of the cursor of the accumulated histogram.

Syntax :ANALysis:AHISTogram<x>:MEASure:CURSor:  
CALCulation?  
<x> = 1 or 2

Example :ANALYSIS:AHISTOGRAM1:MEASURE:CURSOR:  
CALCULATION? -> :ANALYSIS:AHISTOGRAM1:  
MEASURE:CURSOR:  
CALCULATION:DEFINE1 "C1";DEFINE2 "C2";  
DEFINE3 "C1";DEFINE4 "C2";STATE1 1;  
STATE2 1;STATE3 1;STATE4 1

**:ANALysis:AHISTogram<x>:MEASure:  
CURSor:CALCulation:ALL**

Function Turns ON/OFF all calculation items of the cursor of the accumulated histogram.

Syntax :ANALysis:AHISTogram<x>:MEASure:CURSor:  
CALCulation:ALL {<Boolean>}  
<x> = 1 or 2

Example :ANALYSIS:AHISTOGRAM1:MEASURE:CURSOR:  
CALCULATION:ALL ON

### 5.3 ANALYSIS Group

#### **:ANALYSIS:AHISTogram<x>:MEASURE: CURSOR:CALCulation:DEFine<x>**

**Function** Sets the equation of the calculation item of the cursor of the accumulated histogram or queries the current setting.

**Syntax** :ANALYSIS:AHISTogram<x>:MEASURE:CURSOR:  
CALCulation:DEFine<x> {<String>}  
:ANALYSIS:AHISTogram<x>:MEASURE:CURSOR:  
CALCulation:DEFine<x>?  
<x> of AHISTogram<x> = 1 or 2  
<x> of DEFine<x> = 1 to 4  
<String> = Up to 128 characters

**Example** :ANALYSIS:AHISTOGRAM1:MEASURE:CURSOR:  
CALCULATION:DEFINE1 "C1"  
:ANALYSIS:AHISTOGRAM1:MEASURE:CURSOR:  
CALCULATION:DEFINE1? -> :ANALYSIS:  
AHISTOGRAM1:MEASURE:CURSOR:  
CALCULATION:DEFINE1 "C1"

#### **:ANALYSIS:AHISTogram<x>:MEASURE: CURSOR:CALCulation:STATE<x>**

**Function** Turns ON/OFF the calculation item of the cursor of the accumulated histogram or queries the current setting.

**Syntax** :ANALYSIS:AHISTogram<x>:MEASURE:CURSOR:  
CALCulation:STATE<x> {<Boolean>}  
:ANALYSIS:AHISTogram<x>:MEASURE:CURSOR:  
CALCulation:STATE<x>?  
<x> of AHISTogram<x> = 1 or 2  
<x> of STATE<x> = 1 to 4

**Example** :ANALYSIS:AHISTOGRAM1:MEASURE:CURSOR:  
CALCULATION:STATE1 ON  
:ANALYSIS:AHISTOGRAM1:MEASURE:CURSOR:  
CALCULATION:STATE1? -> :ANALYSIS:  
AHISTOGRAM1:MEASURE:CURSOR:  
CALCULATION:STATE1 1

#### **:ANALYSIS:AHISTogram<x>:MEASURE: CURSOR:CALCulation:VALue<x>?**

**Function** Queries the measured value of the calculation item of the cursor of the accumulated histogram.

**Syntax** :ANALYSIS:AHISTogram<x>:MEASURE:CURSOR:  
CALCulation:VALue<x>?  
<x> of AHISTogram<x> = 1 or 2  
<x> of VALue<x> = 1 to 4

**Example** :ANALYSIS:AHISTOGRAM1:MEASURE:CURSOR:  
CALCULATION:VALUE1? -> :ANALYSIS:  
AHISTOGRAM1:MEASURE:CURSOR:  
CALCULATION:VALUE1 1.000E+00

#### **:ANALYSIS:AHISTogram<x>:MEASURE: CURSOR:HLINKage**

**Function** Turns ON/OFF the horizontal cursor link of the accumulated histogram or queries the current setting.

**Syntax** :ANALYSIS:AHISTogram<x>:MEASURE:CURSOR:  
HLINKage {<Boolean>}  
:ANALYSIS:AHISTogram<x>:MEASURE:CURSOR:  
HLINKage?  
<x> = 1 or 2

**Example** :ANALYSIS:AHISTOGRAM1:MEASURE:CURSOR:  
HLINKAGE ON  
:ANALYSIS:AHISTOGRAM1:MEASURE:CURSOR:  
HLINKAGE? -> :ANALYSIS:AHISTOGRAM1:  
MEASURE:CURSOR:HLINKAGE 1

#### **:ANALYSIS:AHISTogram<x>:MEASURE: CURSOR:HPOSITION<x>**

**Function** Sets the horizontal cursor position of the accumulated histogram or queries the current setting.

**Syntax** :ANALYSIS:AHISTogram<x>:MEASURE:CURSOR:  
HPOSITION<x> {<Nrf>}  
:ANALYSIS:AHISTogram<x>:MEASURE:CURSOR:  
HPOSITION<x>?  
<x> of AHISTogram<x> = 1 or 2  
<x> of HPOSITION<x> = 1 or 2  
<Nrf> = -5 to 5 div

**Example** :ANALYSIS:AHISTOGRAM1:MEASURE:CURSOR:  
HPOSITION1 1  
:ANALYSIS:AHISTOGRAM1:MEASURE:CURSOR:  
HPOSITION1? -> :ANALYSIS:AHISTOGRAM1:  
MEASURE:CURSOR:  
HPOSITION1 1.000E+00

#### **:ANALYSIS:AHISTogram<x>:MEASURE: CURSOR:VLINKage**

**Function** Turns ON/OFF the vertical cursor link of the accumulated histogram or queries the current setting.

**Syntax** :ANALYSIS:AHISTogram<x>:MEASURE:CURSOR:  
VLINKage {<Boolean>}  
:ANALYSIS:AHISTogram<x>:MEASURE:CURSOR:  
VLINKage?  
<x> = 1 or 2

**Example** :ANALYSIS:AHISTOGRAM1:MEASURE:CURSOR:  
VLINKAGE ON  
:ANALYSIS:AHISTOGRAM1:MEASURE:CURSOR:  
VLINKAGE? -> :ANALYSIS:AHISTOGRAM1:  
MEASURE:CURSOR:VLINKAGE 1



**:ANALysis:AHIStogram<x>:MEASure:****CURSor:VPOSition<x>**

**Function** Sets the vertical cursor position of the accumulated histogram or queries the current setting.

**Syntax** :ANALysis:AHIStogram<x>:MEASure:CURSor:  
VPOSition<x> {<NRf>}  
:ANALysis:AHIStogram<x>:MEASure:CURSor:  
VPOSition<x>?

<x> of AHIStogram<x> = 1 or 2

<x> of VPOSition<x> = 1 or 2

<NRf> = -4 to 4 (div)

**Example** :ANALYSIS:AHISTOGRAM1:MEASURE:CURSOR:  
VPOSITION1 1  
:ANALYSIS:AHISTOGRAM1:MEASURE:CURSOR:  
VPOSITION1? -> :ANALYSIS:AHISTOGRAM1:  
MEASURE:CURSOR:  
VPOSITION1 1.000E+00

**:ANALysis:AHIStogram<x>:MEASure:MODE**

**Function** Sets the automated measurement mode of the accumulated histogram or queries the current setting.

**Syntax** :ANALysis:AHIStogram<x>:MEASure:MODE  
{CURSor|OFF|PARAMeter}  
:ANALysis:AHIStogram<x>:MEASure:MODE?  
<x> = 1 or 2

**Example** :ANALYSIS:AHISTOGRAM1:MEASURE:  
MODE CURSOR  
:ANALYSIS:AHISTOGRAM1:MEASURE:MODE?  
-> :ANALYSIS:AHISTOGRAM1:MEASURE:  
MODE CURSOR

**:ANALysis:AHIStogram<x>:MEASure:****PARAMeter?**

**Function** Queries all settings related to the automated measurement of waveform parameters of the accumulated histogram.

**Syntax** :ANALysis:AHIStogram<x>:MEASure:  
PARAMeter?  
<x> = 1 or 2

**Example** :ANALYSIS:AHISTOGRAM1:MEASURE:  
PARAMETER? -> :ANALYSIS:AHISTOGRAM1:  
MEASURE:PARAMETER:AREA1:MAXIMUM:  
STATE 1;:ANALYSIS:AHISTOGRAM1:MEASURE:  
PARAMETER:AREA1:MEAN:STATE 1;:ANALYSIS:  
AHISTOGRAM1:MEASURE:PARAMETER:AREA1:  
MEDIAN:STATE 1;:ANALYSIS:AHISTOGRAM1:  
MEASURE:PARAMETER:AREA1:MINIMUM:  
STATE 1;:ANALYSIS:AHISTOGRAM1:MEASURE:  
PARAMETER:AREA1:PEAK:STATE 1;:ANALYSIS:  
AHISTOGRAM1:MEASURE:PARAMETER:AREA1:  
RMS:STATE 1;:ANALYSIS:AHISTOGRAM1:  
MEASURE:PARAMETER:AREA1:SD2INTEG:  
STATE 1;:ANALYSIS:AHISTOGRAM1:MEASURE:  
PARAMETER:AREA1:SD3INTEG:STATE 1;:  
ANALYSIS:AHISTOGRAM1:MEASURE:PARAMETER:  
AREA1:SDEVIATION:STATE 1;:ANALYSIS:  
AHISTOGRAM1:MEASURE:PARAMETER:AREA1:  
SDINTEG:STATE 1;:ANALYSIS:AHISTOGRAM1:  
MEASURE:PARAMETER:AREA2:MAXIMUM:  
STATE 1;:ANALYSIS:AHISTOGRAM1:MEASURE:  
PARAMETER:AREA2:MEAN:STATE 1;:ANALYSIS:  
AHISTOGRAM1:MEASURE:PARAMETER:AREA2:  
MEDIAN:STATE 1;:ANALYSIS:AHISTOGRAM1:  
MEASURE:PARAMETER:AREA2:MINIMUM:  
STATE 1;:ANALYSIS:AHISTOGRAM1:MEASURE:  
PARAMETER:AREA2:PEAK:STATE 1;:ANALYSIS:  
AHISTOGRAM1:MEASURE:PARAMETER:AREA2:  
RMS:STATE 1;:ANALYSIS:AHISTOGRAM1:  
MEASURE:PARAMETER:AREA2:SD2INTEG:  
STATE 1;:ANALYSIS:AHISTOGRAM1:MEASURE:  
PARAMETER:AREA2:SD3INTEG:STATE 1;:  
ANALYSIS:AHISTOGRAM1:MEASURE:PARAMETER:  
AREA2:SDEVIATION:STATE 1;:ANALYSIS:  
AHISTOGRAM1:MEASURE:PARAMETER:AREA2:  
SDINTEG:STATE 1.....

### 5.3 ANALYSIS Group

**:ANALYSIS:AHISTogram<x>:MEASURE:  
PARAMETER:AREA<x>?**

Function Queries all settings related to the area of the accumulated histogram.

Syntax :ANALYSIS:AHISTogram<x>:MEASURE:  
PARAMETER:AREA<x>?  
<x> of AHISTogram<x> = 1 or 2  
<x> of AREA<x> = 1 or 2

Example :ANALYSIS:AHISTOGRAM1:MEASURE:  
PARAMETER:AREA1? -> :ANALYSIS:  
AHISTOGRAM1:MEASURE:  
PARAMETER:AREA1:MAXIMUM:STATE 1;;  
ANALYSIS:AHISTOGRAM1:MEASURE:PARAMETER:  
AREA1:MEAN:STATE 1;;ANALYSIS:  
AHISTOGRAM1:MEASURE:PARAMETER:AREA1:  
MEDIAN:STATE 1;;ANALYSIS:AHISTOGRAM1:  
MEASURE:PARAMETER:AREA1:MINIMUM:  
STATE 1;;ANALYSIS:AHISTOGRAM1:MEASURE:  
PARAMETER:AREA1:PEAK:STATE 1;;ANALYSIS:  
AHISTOGRAM1:MEASURE:PARAMETER:AREA1:  
RMS:STATE 1;;ANALYSIS:AHISTOGRAM1:  
MEASURE:PARAMETER:AREA1:SD2INTEG:  
STATE 1;;ANALYSIS:AHISTOGRAM1:MEASURE:  
PARAMETER:AREA1:SD3INTEG:STATE 1;;  
ANALYSIS:AHISTOGRAM1:MEASURE:PARAMETER:  
AREA1:SDEVIATION:STATE 1;;ANALYSIS:  
AHISTOGRAM1:MEASURE:PARAMETER:AREA1:  
SDINTEG:STATE 1

**:ANALYSIS:AHISTogram<x>:MEASURE:  
PARAMETER:AREA<x>:ALL**

Function Turns ON/OFF all waveform parameters of the accumulated histogram.

Syntax :ANALYSIS:AHISTogram<x>:MEASURE:  
PARAMETER:AREA<x>:ALL {<Boolean>}  
<x> of AHISTogram<x> = 1 or 2  
<x> of AREA<x> = 1 or 2

Example :ANALYSIS:AHISTOGRAM1:MEASURE:  
PARAMETER:AREA1:ALL ON

**:ANALYSIS:AHISTogram<x>:MEASURE:  
PARAMETER:AREA<x>:<Parameter>?**

Function Queries all settings related to the waveform parameter of the accumulated histogram.

Syntax :ANALYSIS:AHISTogram<x>:MEASURE:  
PARAMETER:AREA<x>:<Parameter>?  
<x> of AHISTogram<x> = 1 or 2  
<x> of AREA<x> = 1 or 2

<Parameter> = {MAXimum|MEAN|MEDian|  
MINimum|PEAK|RMS|SD2integ|SD3integ|  
SDEVIation|SDINteg}

Example (The following is an example for the maximum value of area 1.)

```
:ANALYSIS:AHISTOGRAM1:MEASURE:
PARAMETER:AREA1:MAXIMUM? -> :ANALYSIS:
AHISTOGRAM1:MEASURE:
PARAMETER:AREA1:MAXIMUM:STATE 1
```

**:ANALYSIS:AHISTogram<x>:MEASURE:  
PARAMETER:AREA<x>:<Parameter>:STATE**

Function Turns ON/OFF the waveform parameter of the accumulated histogram or queries the current setting.

Syntax :ANALYSIS:AHISTogram<x>:MEASURE:  
PARAMETER:AREA<x>:<Parameter>:  
STATE {<Boolean>}  
:ANALYSIS:AHISTogram<x>:MEASURE:  
PARAMETER:AREA<x>:<Parameter>:STATE?  
<x> of AHISTogram<x> = 1 or 2  
<x> of AREA<x> = 1 or 2

<Parameter> = {MAXimum|MEAN|MEDian|  
MINimum|PEAK|RMS|SD2integ|SD3integ|  
SDEVIation|SDINteg}

Example (The following is an example for the maximum value of area 1.)

```
:ANALYSIS:AHISTOGRAM1:MEASURE:
PARAMETER:AREA1:MAXIMUM:STATE ON
:ANALYSIS:AHISTOGRAM1:MEASURE:
PARAMETER:AREA1:MAXIMUM:STATE?
-> :ANALYSIS:AHISTOGRAM1:MEASURE:
PARAMETER:AREA1:MAXIMUM:STATE 1
```

**:ANALysis:AHIStoqram<x>:MEASure:  
PARAmeter:AREA<x>:<Parameter>:VALue?**

**Function** Queries the automated measured value of the waveform parameter of the accumulated histogram.

**Syntax** :ANALysis:AHIStoqram<x>:MEASure:  
PARAmeter:AREA<x>:<Parameter>:VALue?  
<x> of AHIStoqram<x> = 1 or 2  
<x> of AREA<x> = 1 or 2  
<Parameter> = {MAXimum|MEAN|MEDIan|  
MINimum|PEAK|RMS|SD2integ|SD3integ|  
SDEVIation|SDINteq}

**Example** (The following is an example for the maximum value of area 1.)

```
:ANALYSIS:AHISTOGRAM1:MEASURE:
PARAMETER:AREA1:MAXIMUM:VALUE?
-> :ANALYSIS:AHISTOGRAM1:MEASURE:
PARAMETER:AREA1:MAXIMUM:VALUE 1.000E+00
```

**:ANALysis:AHIStoqram<x>:MEASure:  
PARAmeter:CALCulation?**

**Function** Queries all settings related to the calculation items of waveform parameters of the accumulated histogram.

**Syntax** :ANALysis:AHIStoqram<x>:MEASure:  
PARAmeter:CALCulation?  
<x> = 1 or 2

**Example** :ANALYSIS:AHISTOGRAM1:MEASURE:  
PARAMETER:CALCULATION? -> :ANALYSIS:  
AHISTOGRAM1:MEASURE:  
PARAMETER:CALCULATION:DEFINE1 "MEAN";  
DEFINE2 "MAX";DEFINE3 "MIN";  
DEFINE4 "PEAK";STATE1 1;STATE2 1;  
STATE3 1;STATE4 1

**:ANALysis:AHIStoqram<x>:MEASure:  
PARAmeter:CALCulation:ALL**

**Function** Turns ON/OFF all calculation items of the waveform parameters of the accumulated histogram.

**Syntax** :ANALysis:AHIStoqram<x>:MEASure:  
PARAmeter:CALCulation:ALL {<Boolean>}  
<x> = 1 or 2

**Example** :ANALYSIS:AHISTOGRAM1:MEASURE:  
PARAMETER:CALCULATION:ALL ON

**:ANALysis:AHIStoqram<x>:MEASure:  
PARAmeter:CALCulation:DEFine<x>**

**Function** Sets the equation of the calculation items of the waveform parameter of the accumulated histogram or queries the current setting.

**Syntax** :ANALysis:AHIStoqram<x>:MEASure:  
PARAmeter:CALCulation:  
DEFine<x> {<String>}  
<x> of AHIStoqram<x> = 1 or 2  
<x> of DEFine<x> = 1 to 4  
<String> = Up to 128 characters

**Example** :ANALYSIS:AHISTOGRAM1:MEASURE:  
PARAMETER:CALCULATION:DEFINE1 "MEAN"  
:ANALYSIS:AHISTOGRAM1:MEASURE:  
PARAMETER:CALCULATION:DEFINE1?  
-> :ANALYSIS:AHISTOGRAM1:MEASURE:  
PARAMETER:CALCULATION:DEFINE1 "MEAN"

**:ANALysis:AHIStoqram<x>:MEASure:  
PARAmeter:CALCulation:STATe<x>**

**Function** Turns ON/OFF the calculation items of the waveform parameter of the accumulated histogram or queries the current setting.

**Syntax** :ANALysis:AHIStoqram<x>:MEASure:  
PARAmeter:CALCulation:STATe<x>  
{<Boolean>}  
<x> of AHIStoqram<x> = 1 or 2  
<x> of STATe<x> = 1 to 4

**Example** :ANALYSIS:AHISTOGRAM1:MEASURE:  
PARAMETER:CALCULATION:STATE1 ON  
:ANALYSIS:AHISTOGRAM1:MEASURE:  
PARAMETER:CALCULATION:STATE1?  
-> :ANALYSIS:AHISTOGRAM1:MEASURE:  
PARAMETER:CALCULATION:STATE1 1

**:ANALysis:AHIStoqram<x>:MEASure:  
PARAmeter:CALCulation:VALue<x>?**

**Function** Queries the automated measured value of the calculation items of the waveform parameter of the accumulated histogram.

**Syntax** :ANALysis:AHIStoqram<x>:MEASure:  
PARAmeter:CALCulation:VALue<x>?  
<x> of AHIStoqram<x> = 1 or 2  
<x> of VALue<x> = 1 to 4

**Example** :ANALYSIS:AHISTOGRAM1:MEASURE:  
PARAMETER:CALCULATION:VALUE1?  
-> :ANALYSIS:AHISTOGRAM1:MEASURE:  
PARAMETER:CALCULATION:VALUE1 1.000E+00

### 5.3 ANALysis Group

#### **:ANALysis:AHISTogram<x>:MEASure:**

##### **PARAmeter:HRANge<x>**

**Function** Sets the horizontal range of the waveform parameter of the accumulated histogram or queries the current setting.

**Syntax** :ANALysis:AHISTogram<x>:MEASure:  
PARAmeter:HRANge<x> {<NRf>,<NRf>}  
:ANALysis:AHISTogram<x>:MEASure:  
PARAmeter:HRANge<x>?  
<x> of AHISTogram<x> = 1 or 2  
<x> of HRANge<x> = 1 or 2  
<NRf> = -5 to 5 div

**Example** :ANALYSIS:AHISTOGRAM1:MEASURE:  
PARAMETER:HRANGE1 1,2  
:ANALYSIS:AHISTOGRAM1:MEASURE:  
PARAMETER:HRANGE1? -> :ANALYSIS:  
AHISTOGRAM1:MEASURE:  
PARAMETER:HRANGE1 2.000E+00,1.000E+00

#### **:ANALysis:AHISTogram<x>:MEASure:**

##### **PARAmeter:VRANge<x>**

**Function** Sets the vertical range of the waveform parameter of the accumulated histogram or queries the current setting.

**Syntax** :ANALysis:AHISTogram<x>:MEASure:  
PARAmeter:VRANge<x> {<NRf>,<NRf>}  
:ANALysis:AHISTogram<x>:MEASure:  
PARAmeter:VRANge<x>?  
<x> of AHISTogram<x> = 1 or 2  
<x> of VRANge<x> = 1 or 2  
<NRf> = -4 to 4 (div)

**Example** :ANALYSIS:AHISTOGRAM1:MEASURE:  
PARAMETER:VRANGE1 1,2  
:ANALYSIS:AHISTOGRAM1:MEASURE:  
PARAMETER:VRANGE1? -> :ANALYSIS:  
AHISTOGRAM1:MEASURE:  
PARAMETER:VRANGE1 2.000E+00,1.000E+00

#### **:ANALysis:AHISTogram<x>:MODE**

**Function** Sets the accumulated histogram mode or queries the current setting.

**Syntax** :ANALysis:AHISTogram<x>:MODE  
{HORizontal|VERTical}  
:ANALysis:AHISTogram<x>:MODE?  
<x> = 1 or 2

**Example** :ANALYSIS:AHISTOGRAM1:MODE HORIZONTAL  
:ANALYSIS:AHISTOGRAM1:MODE?  
-> :ANALYSIS:AHISTOGRAM1:MODE HORIZONTAL

#### **:ANALysis:AHISTogram<x>:TRACe**

**Function** Sets the source trace of the accumulated histogram or queries the current setting.

**Syntax** :ANALysis:AHISTogram<x>:TRACe {<NRf>}  
:ANALysis:AHISTogram<x>:TRACe?  
<x> = 1 or 2  
<NRf> = 1 to 8

**Example** :ANALYSIS:AHISTOGRAM1:TRACE 1  
:ANALYSIS:AHISTOGRAM1:TRACE?  
-> :ANALYSIS:AHISTOGRAM1:TRACE 1

#### **:ANALysis:AHISTogram<x>:VERTical**

**Function** Sets the vertical range of the accumulated histogram or queries the current setting.

**Syntax** :ANALysis:AHISTogram<x>:VERTical  
{<NRf>,<NRf>}  
:ANALysis:AHISTogram<x>:VERTical?  
<x> = 1 or 2  
<NRf> = -4 to 4 (div)

**Example** :ANALYSIS:AHISTOGRAM1:VERTICAL 1,2  
:ANALYSIS:AHISTOGRAM1:VERTICAL?  
-> :ANALYSIS:AHISTOGRAM1:  
VERTICAL 2.000E+00,1.000E+00

#### **:ANALysis:AHISTogram<x>:WINDow**

**Function** Sets the measurement target window of the accumulated histogram or queries the current setting.

**Syntax** :ANALysis:AHISTogram<x>:WINDow {MAIN|  
Z1|Z2}  
:ANALysis:AHISTogram<x>:WINDow?  
<x> = 1 or 2

**Example** :ANALYSIS:AHISTOGRAM1:WINDOW MAIN  
:ANALYSIS:AHISTOGRAM1:WINDOW?  
-> :ANALYSIS:AHISTOGRAM1:WINDOW MAIN

#### **:ANALysis:DISPlay<x>**

**Function** Turns ON/OFF the analysis function display or queries the current setting.

**Syntax** :ANALysis:DISPlay<x> {<Boolean>}  
:ANALysis:DISPlay<x>?  
<x> = 1 or 2

**Example** :ANALYSIS:DISPLAY1 ON  
:ANALYSIS:DISPLAY1? -> :ANALYSIS:  
DISPLAY1 1

**:ANALysis:FFT<x>?**

**Function** Queries all settings related to the FFT computation function.

**Syntax** :ANALysis:FFT<x>?  
<x> = 1 or 2

**Example** :ANALYSIS:FFT1? -> :ANALYSIS:FFT1:  
HORIZONTAL:CPAN:  
CENTER 1.000E+00;SPAN1.000E+00;  
:ANALYSIS:FFT1:HORIZONTAL:  
LRIGHT:RANGE2.000E+00,1.000E+00;  
:ANALYSIS:FFT1:HORIZONTAL:MODE AUTO;  
:ANALYSIS:FFT1:IPART 1;LENGTH 2500;  
MAXHOLD 1;MEASURE:MARKer:BASIC:  
DFREQUENCY:STATE1;:ANALYSIS:FFT1:  
MEASURE:MARKER:BASIC:DV:STATE 1;  
:ANALYSIS:FFT1:MEASURE:MARKER:BASIC:  
FREQUENCY1:STATE 1;  
:ANALYSIS:FFT1:MEASURE:MARKER:BASIC:  
FREQUENCY2:STATE 1;  
:ANALYSIS:FFT1:MEASURE:MARKER:BASIC:  
POSITION1 1.000E+00;  
:ANALYSIS:FFT1:MEASURE:MARKER:BASIC:  
POSITION2 2.000E+00;V1:STATE 1;  
:ANALYSIS:FFT1:MEASURE:MARKER:BASIC:  
V2:STATE 1;:ANALYSIS:FFT1:MEASURE:  
MARKER:CALCULATION:DEFINE1 "V(F1)";  
DEFINE2 "V(F2)";DEFINE3 "V(F1)";  
DEFINE4 "V(F2)";STATE1 1;STATE2 1;  
STATE3 1;STATE4 1;  
:ANALYSIS:FFT1:MEASURE:MODE MARKER;  
PEAK:BASIC:DFREQUENCY:STATE 1;  
:ANALYSIS:FFT1:MEASURE:PEAK:BASIC:  
DV:STATE 1;  
:ANALYSIS:FFT1:MEASURE:PEAK:BASIC:  
FREQUENCY1:STATE 1;  
:ANALYSIS:FFT1:MEASURE:PEAK:BASIC:  
FREQUENCY2:STATE 1;  
:ANALYSIS:FFT1:MEASURE:PEAK:BASIC:  
RANGE1 1.000E+00,0.000E+00;  
RANGE2 1.000E+00,0.000E+00;V1:  
STATE 1;  
:ANALYSIS:FFT1:MEASURE:PEAK:BASIC:V2:  
STATE 1.....

**:ANALysis:FFT<x>:HORizontal?**

**Function** Queries all settings related the horizontal axis of the FFT computation.

**Syntax** :ANALysis:FFT<x>:HORizontal?  
<x> = 1 or 2

**Example** :ANALYSIS:FFT1:HORIZONTAL?  
-> :ANALYSIS:FFT1:HORIZONTAL:CPAN:  
CENTER 1.000E+00;SPAN 1.000E+00;:  
ANALYSIS:FFT1:HORIZONTAL:LRIGHT:  
RANGE 2.000E+00,1.000E+00;:ANALYSIS:  
FFT1:HORIZONTAL:MODE AUTO

**:ANALysis:FFT<x>:HORizontal:CPan?**

**Function** Queries all settings related to the center and span of the horizontal axis of the FFT computation.

**Syntax** :ANALysis:FFT<x>:HORizontal:CPan?  
<x> = 1 or 2

**Example** :ANALYSIS:FFT1:HORIZONTAL:CPAN?  
-> :ANALYSIS:FFT1:HORIZONTAL:CPAN:  
CENTER 1.000E+00;SPAN 1.000E+00

**:ANALysis:FFT<x>:HORizontal:CPan:CENTer**

**Function** Sets the horizontal center of the FFT computation or queries the current setting.

**Syntax** :ANALysis:FFT<x>:HORizontal:CPan:  
CENTer {<Frequency>}  
:ANALysis:FFT<x>:HORizontal:CPan:  
CENTer?  
<x> = 1 or 2  
<Frequency> = 0 to 250 G (Hz)

**Example** :ANALYSIS:FFT1:HORIZONTAL:CPAN:  
CENTER 1HZ  
:ANALYSIS:FFT1:HORIZONTAL:CPAN:CENTER?  
-> :ANALYSIS:FFT1:HORIZONTAL:CPAN:  
CENTER 1.000E+00

**:ANALysis:FFT<x>:HORizontal:CPan:SPAN**

**Function** Sets the horizontal span of the FFT computation or queries the current setting.

**Syntax** :ANALysis:FFT<x>:HORizontal:CPan:  
SPAN {<Frequency>}  
:ANALysis:FFT<x>:HORizontal:CPan:SPAN?  
<x> = 1 or 2  
<Frequency> = 0 to 250 G (Hz)

**Example** :ANALYSIS:FFT1:HORIZONTAL:CPAN:  
SPAN 1HZ  
:ANALYSIS:FFT1:HORIZONTAL:CPAN:SPAN?  
-> :ANALYSIS:FFT1:HORIZONTAL:CPAN:  
SPAN 1.000E+00

**:ANALysis:FFT<x>:HORizontal:LRIGHT?**

**Function** Queries all settings related the left and right edges of the horizontal axis of the FFT computation.

**Syntax** :ANALysis:FFT<x>:HORizontal:LRIGHT?  
<x> = 1 or 2

**Example** :ANALYSIS:FFT1:HORIZONTAL:LRIGHT?  
-> :ANALYSIS:FFT1:HORIZONTAL:LRIGHT:  
RANGE 2.000E+00,1.000E+00

### 5.3 ANALysis Group

#### **:ANALysis:FFT<x>:HORizontal:LRIGHT:**

##### **RANGE**

**Function** Sets the range of the horizontal left and right edges of the FFT computation or queries the current setting.

**Syntax** :ANALysis:FFT<x>:HORizontal:LRIGHT:  
RANGE {<Frequency>,<Frequency>}  
:ANALysis:FFT<x>:HORizontal:LRIGHT:  
RANGE?  
<x> = 1 or 2  
<Frequency> = 0 to 250 G (Hz)

**Example** :ANALYSIS:FFT1:HORIZONTAL:LRIGHT:  
RANGE 1HZ,2HZ  
:ANALYSIS:FFT1:HORIZONTAL:LRIGHT:RANGE?  
-> :ANALYSIS:FFT1:HORIZONTAL:LRIGHT:  
RANGE 2.000E+00,1.000E+00

#### **:ANALysis:FFT<x>:HORizontal:MODE**

**Function** Sets the horizontal mode of the FFT computation or queries the current setting.

**Syntax** :ANALysis:FFT<x>:HORizontal:MODE {AUTO|  
CSPan|LRIGHT}  
:ANALysis:FFT<x>:HORizontal:MODE?  
<x> = 1 or 2

**Example** :ANALYSIS:FFT1:HORIZONTAL:MODE AUTO  
:ANALYSIS:FFT1:HORIZONTAL:MODE?  
-> :ANALYSIS:FFT1:HORIZONTAL:MODE AUTO

#### **:ANALysis:FFT<x>:IPART (Imag Part)**

**Function** Sets the source trace of the imaginary part of the FFT computation or queries the current setting.

**Syntax** :ANALysis:FFT<x>:IPART {<Nrf>|DONTcare}  
:ANALysis:FFT<x>:IPART?  
<x> = 1 or 2  
<Nrf> = 1 to 8

**Example** :ANALYSIS:FFT1:IPART 1  
:ANALYSIS:FFT1:IPART? -> :ANALYSIS:  
FFT1:IPART 1

#### **:ANALysis:FFT<x>:LENGth**

**Function** Sets the number of FFT points of the FFT computation or queries the current setting.

**Syntax** :ANALysis:FFT<x>:LENGth {<Nrf>}  
:ANALysis:FFT<x>:LENGth?  
<x> = 1 or 2  
<Nrf> = 2500, 6250, 12500, 25000, 62500, 125000,  
or 250000

**Example** :ANALYSIS:FFT1:LENGTH 2500  
:ANALYSIS:FFT1:LENGTH? -> :ANALYSIS:  
FFT1:LENGTH 2500

#### **:ANALysis:FFT<x>:MAXHold**

**Function** Turns ON/OFF the maximum value hold function of the FFT computation or queries the current setting.

**Syntax** :ANALysis:FFT<x>:MAXHold {<Boolean>}  
:ANALysis:FFT<x>:MAXHold?  
<x> = 1 or 2

**Example** :ANALYSIS:FFT1:MAXHOLD ON  
:ANALYSIS:FFT1:MAXHOLD? -> :ANALYSIS:  
FFT1:MAXHOLD 1

#### **:ANALysis:FFT<x>:MEASure?**

**Function** Queries all settings related to the automated measurement of the FFT computation.

**Syntax** :ANALysis:FFT<x>:MEASure?  
<x> = 1 or 2

**Example** :ANALYSIS:FFT1:MEASURE? -> :ANALYSIS:  
FFT1:MEASURE:MARKer:BASIC:  
DFREQUENCY:STATE 1;  
:ANALYSIS:FFT1:MEASURE:MARKer:BASIC:  
DV:STATE 1;  
:ANALYSIS:FFT1:MEASURE:MARKer:BASIC:  
FREQUENCY1:STATE 1;  
:ANALYSIS:FFT1:MEASURE:MARKer:BASIC:  
FREQUENCY2:STATE 1;  
:ANALYSIS:FFT1:MEASURE:MARKer:BASIC:  
POSITION1 1.000E+00;  
:ANALYSIS:FFT1:MEASURE:  
MARKer:BASIC:POSITION2 2.000E+00;V1:  
STATE 1;  
:ANALYSIS:FFT1:MEASURE:MARKer:BASIC:  
V2:STATE 1;  
:ANALYSIS:FFT1:MEASURE:MARKer:  
CALCULATION:DEFINE1 "V(F1)";DEFINE2  
"V(F2)";DEFINE3 "V(F1)";DEFINE4  
"V(F2)";STATE1 1;STATE2 1;STATE3  
1;STATE4 1;  
:ANALYSIS:FFT1:MEASURE:MODE  
MARKer;PEAK:BASIC:DFREQUENCY:STATE 1;  
:ANALYSIS:FFT1:MEASURE:PEAK:BASIC:  
DV:STATE 1;  
:ANALYSIS:FFT1:MEASURE:PEAK:BASIC:  
FREQUENCY1:STATE 1;  
:ANALYSIS:FFT1:MEASURE:PEAK:BASIC:  
FREQUENCY2:STATE 1;  
:ANALYSIS:FFT1:MEASURE:PEAK:BASIC:  
RANGE1 1.000E+00,0.000E+00;RANGE2  
1.000E+00,0.000E+00;V1:STATE 1;  
:ANALYSIS:FFT1:MEASURE:PEAK:BASIC:  
V2:STATE 1;  
:ANALYSIS:FFT1:MEASURE:PEAK:  
CALCULATION:DEFINE1 "V(P1)";DEFINE2  
"V(P2)";DEFINE3 "V(P1)";DEFINE4  
"V(P2)";STATE1 1;STATE2 1;STATE3  
1;STATE4 1

**:ANALysis:FFT<x>:MEASure:MARKer?**

**Function** Queries all settings related to the marker cursor measurement of the FFT computation.

**Syntax** :ANALysis:FFT<x>:MEASure:MARKer?  
<x> = 1 or 2

**Example**

```
:ANALYSIS:FFT1:MEASURE:MARKER?
-> :ANALYSIS:FFT1:MEASURE:MARKER:BASIC:
DFREQUENCY:STATE 1;
:ANALYSIS:FFT1:MEASURE:MARKER:BASIC:
DV:STATE 1;
:ANALYSIS:FFT1:MEASURE:MARKER:BASIC:
FREQUENCY1:STATE 1;
:ANALYSIS:FFT1:MEASURE:MARKER:BASIC:
FREQUENCY2:STATE 1;
:ANALYSIS:FFT1:MEASURE:MARKER:BASIC:
POSITION1 1.000E+00;
:ANALYSIS:FFT1:MEASURE:MARKER:BASIC:
POSITION2 2.000E+00;V1:STATE 1;
:ANALYSIS:FFT1:MEASURE:MARKER:BASIC:
V2:STATE 1;
:ANALYSIS:FFT1:MEASURE:MARKER:
CALCULATION:DEFINE1 "V(F1)";DEFINE2
"V(F2)";DEFINE3 "V(F1)";DEFINE4
"V(F2)";STATE1 1;STATE2 1;STATE3
1;STATE4 1
```

**:ANALysis:FFT<x>:MEASure:****MARKer[:BASic]?**

**Function** Queries all settings related to basic items of the marker cursor of the FFT computation.

**Syntax** :ANALysis:FFT<x>:MEASure:  
MARKer:BASic?

**Example**

```
:ANALYSIS:FFT1:MEASURE:MARKER:BASIC?
-> :ANALYSIS:FFT1:MEASURE:MARKER:BASIC:
DFREQUENCY:STATE 1;
:ANALYSIS:FFT1:MEASURE:MARKER:
BASIC:DV:STATE 1;
:ANALYSIS:FFT1:MEASURE:MARKER:BASIC:
FREQUENCY1:STATE 1;
:ANALYSIS:FFT1:MEASURE:MARKER:BASIC:
FREQUENCY2:STATE 1;
:ANALYSIS:FFT1:MEASURE:MARKER:
BASIC:POSITION1 1.000E+00;
:ANALYSIS:FFT1:MEASURE:
MARKER:BASIC:POSITION2 2.000E+00;V1:
STATE 1;
:ANALYSIS:FFT1:MEASURE:MARKER:
BASIC:V2:STATE 1
```

**:ANALysis:FFT<x>:MEASure:****MARKer[:BASic]:ALL**

**Function** Turns ON/OFF all basic items of the marker cursor of the FFT computation.

**Syntax** :ANALysis:FFT<x>:MEASure:  
MARKer[:BASic]:ALL {<Boolean>}  
<x> = 1 or 2

**Example** :ANALYSIS:FFT1:MEASURE:MARKER:BASIC:  
ALL ON

**:ANALysis:FFT<x>:MEASure:****MARKer[:BASic]:DFrequency?**

**Function** Queries all settings related to the frequency value between marker cursors of the FFT computation.

**Syntax** :ANALysis:FFT<x>:MEASure:  
MARKer[:BASic]:DFrequency?  
<x> = 1 or 2

**Example** :ANALYSIS:FFT1:MEASURE:MARKER:BASIC:  
DFREQUENCY? -> :ANALYSIS:FFT1:MEASURE:  
MARKER:BASIC:DFREQUENCY:STATE 1

**:ANALysis:FFT<x>:MEASure:****MARKer[:BASic]:DFrequency:STATe**

**Function** Turns ON/OFF the frequency value between marker cursors of the FFT computation or queries the current setting.

**Syntax** :ANALysis:FFT<x>:MEASure:  
MARKer[:BASic]:DFrequency:  
STATe {<Boolean>}  
:ANALysis:FFT<x>:MEASure:  
MARKer[:BASic]:DFrequency:STATe?  
<x> = 1 or 2

**Example** :ANALYSIS:FFT1:MEASURE:MARKER:BASIC:  
DFREQUENCY:STATE ON  
:ANALYSIS:FFT1:MEASURE:MARKER:BASIC:  
DFREQUENCY:STATE?  
->:ANALYSIS:FFT1:MEASURE:MARKER:  
BASIC:DFREQUENCY:STATE 1

**:ANALysis:FFT<x>:MEASure:****MARKer[:BASic]:DFrequency:VALue?**

**Function** Queries the frequency value between marker cursors of the FFT computation.

**Syntax** :ANALysis:FFT<x>:MEASure:  
MARKer[:BASic]:DFrequency:VALue?  
<x> = 1 or 2

**Example** :ANALYSIS:FFT1:MEASURE:MARKER:BASIC:  
DFREQUENCY:VALUE? -> :ANALYSIS:FFT1:  
MEASURE:MARKER:BASIC:  
DFREQUENCY:VALUE 1.000E+00

### 5.3 ANALYSIS Group

#### **:ANALysis:FFT<x>:MEASure:**

##### **MARKer[:BASic]:DV?**

Function Queries all settings related to the power value between marker cursors of the FFT computation.

Syntax :ANALysis:FFT<x>:MEASure:  
MARKer[:BASic]:DV?  
<x> = 1 or 2

Example :ANALYSIS:FFT1:MEASURE:MARKER:BASIC:DV?  
-> :ANALYSIS:FFT1:MEASURE:MARKER:BASIC:  
DV:STATE 1

#### **:ANALysis:FFT<x>:MEASure:**

##### **MARKer[:BASic]:DV:STATe**

Function Turns ON/OFF the power value between marker cursors of the FFT computation or queries the current setting.

Syntax :ANALysis:FFT<x>:MEASure:  
MARKer[:BASic]:DV:STATe {<Boolean>}  
:ANALysis:FFT<x>:MEASure:  
MARKer[:BASic]:DV:STATe?  
<x> = 1 or 2

Example :ANALYSIS:FFT1:MEASURE:MARKER:BASIC:DV:  
STATE ON  
:ANALYSIS:FFT1:MEASURE:MARKER:BASIC:DV:  
STATE? -> :ANALYSIS:FFT1:MEASURE:  
MARKER:BASIC:DV:STATE 1

#### **:ANALysis:FFT<x>:MEASure:**

##### **MARKer[:BASic]:DV:VALue?**

Function Queries the power value between marker cursors of the FFT computation.

Syntax :ANALysis:FFT<x>:MEASure:  
MARKer[:BASic]:DV:VALue?  
<x> = 1 or 2

Example :ANALYSIS:FFT1:MEASURE:MARKER:BASIC:DV:  
VALUE? -> :ANALYSIS:FFT1:MEASURE:  
MARKER:BASIC:DV:VALUE 1.000E+00

#### **:ANALysis:FFT<x>:MEASure:**

##### **MARKer[:BASic]:FREQuency<x>?**

Function Queries all settings related to the frequency value of the marker cursor of the FFT computation.

Syntax :ANALysis:FFT<x>:MEASure:  
MARKer[:BASic]:FREQuency<x>?  
<x> of FFT<x> = 1 or 2  
<x> of FREQuency<x> = 1 or 2

Example :ANALYSIS:FFT1:MEASURE:MARKER:BASIC:  
FREQUENCY1? -> :ANALYSIS:FFT1:MEASURE:  
MARKER:BASIC:FREQUENCY1:STATE 1

#### **:ANALysis:FFT<x>:MEASure:**

##### **MARKer[:BASic]:FREQuency<x>:STATe**

Function Turns ON/OFF the frequency value of the marker cursor of the FFT computation or queries the current setting.

Syntax :ANALysis:FFT<x>:MEASure:  
MARKer[:BASic]:FREQuency<x>:  
STATe {<Boolean>}  
:ANALysis:FFT<x>:MEASure:  
MARKer[:BASic]:FREQuency<x>:STATe?  
<x> of FFT<x> = 1 or 2  
<x> of FREQuency<x> = 1 or 2

Example :ANALYSIS:FFT1:MEASURE:MARKER:BASIC:  
FREQUENCY1:STATE ON  
:ANALYSIS:FFT1:MEASURE:MARKER:BASIC:  
FREQUENCY1:STATE? -> :ANALYSIS:FFT1:  
MEASURE:MARKER:BASIC:  
FREQUENCY1:STATE 1

#### **:ANALysis:FFT<x>:MEASure:**

##### **MARKer[:BASic]:FREQuency<x>:VALue?**

Function Queries the frequency value of the marker cursor of the FFT computation.

Syntax :ANALysis:FFT<x>:MEASure:  
MARKer[:BASic]:FREQuency<x>:VALue?  
<x> of FFT<x> = 1 or 2  
<x> of FREQuency<x> = 1 or 2

Example :ANALYSIS:FFT1:MEASURE:MARKER:BASIC:  
FREQUENCY1:VALUE? -> :ANALYSIS:FFT1:  
MEASURE:MARKER:BASIC:FREQUENCY1:  
VALUE 1.000E+00

#### **:ANALysis:FFT<x>:MEASure:**

##### **MARKer[:BASic]:POSition<x> {<NRf>}**

Function Sets the marker cursor position of the FFT computation or queries the current setting.

Syntax :ANALysis:FFT<x>:MEASure:  
MARKer[:BASic]:POSition<x> {<NRf>}  
:ANALysis:FFT<x>:MEASure:  
MARKer[:BASic]:POSition<x>?  
<x> of FFT<x> = 1 or 2  
<x> of POSition<x> = 1 or 2  
<NRf> = -5 to 5 (div)

Example :ANALYSIS:FFT1:MEASURE:MARKER:BASIC:  
POSITION1 1  
:ANALYSIS:FFT1:MEASURE:MARKER:BASIC:  
POSITION1? -> :ANALYSIS:FFT1:MEASURE:  
MARKER:BASIC:POSITION1 1.000E+00



**:ANALysis:FFT<x>:MEASure:****MARKer[:BASic]:V<x>?**

Function Queries all settings related to the power value of the marker cursor of the FFT computation.

Syntax :ANALysis:FFT<x>:MEASure:

MARKer[:BASic]:V<x>?

<x> of FFT<x> = 1 or 2

<x> of V<x> = 1 or 2

Example :ANALYSIS:FFT1:MEASURE:MARKER: BASIC:V1?

-> :ANALYSIS:FFT1:MEASURE:MARKER:

BASIC:STATE 1

**:ANALysis:FFT<x>:MEASure:****MARKer[:BASic]:V<x>:STATe**

Function Turns ON/OFF the power value of the marker cursor of the FFT computation or queries the current setting.

Syntax :ANALysis:FFT<x>:MEASure:

MARKer[:BASic]:V<x>:STATe {<Boolean>}

:ANALysis:FFT<x>:MEASure:

MARKer[:BASic]:V<x>:STATe?

<x> of FFT<x> = 1 or 2

<x> of V<x> = 1 or 2

Example :ANALYSIS:FFT1:MEASURE:MARKER: BASIC:V1:

STATE ON

:ANALYSIS:FFT1:MEASURE:MARKER: BASIC:V1:

STATE? -> :ANALYSIS:FFT1:MEASURE:

MARKER: BASIC:V1:STATE 1

**:ANALysis:FFT<x>:MEASure:****MARKer[:BASic]:V<x>:VALue?**

Function Queries the power value of the marker cursor of the FFT computation.

Syntax :ANALysis:FFT<x>:MEASure:

MARKer[:BASic]:V<x>:VALue?

<x> of FFT<x> = 1 or 2

<x> of V<x> = 1 or 2

Example :ANALYSIS:FFT1:MEASURE:MARKER: BASIC:V1:

VALUE? -> :ANALYSIS:FFT1:MEASURE:

MARKER: BASIC:V1:VALUE 1.000E+00

**:ANALysis:FFT<x>:MEASure:MARKer:****CALCulation?**

Function Queries all settings related to calculation items of the marker cursor of the FFT computation.

Syntax :ANALysis:FFT<x>:MEASure:MARKer:

CALCulation?

<x> = 1 or 2

Example :ANALYSIS:FFT1:MEASURE:MARKER:

CALCULATION? -> :ANALYSIS:FFT1:MEASURE:

MARKER:CALCULATION:DEFINE1 "V(F1)";

DEFINE2 "V(F2)";DEFINE3 "V(F1)";D

EFINE4 "V(F2)";STATE1 1;STATE2 1;

STATE3 1;STATE4 1

**:ANALysis:FFT<x>:MEASure:MARKer:****CALCulation:ALL**

Function Turns ON/OFF all calculation items of the marker cursor of the FFT computation.

Syntax :ANALysis:FFT<x>:MEASure:MARKer:

CALCulation:ALL {<Boolean>}

<x> = 1 or 2

Example :ANALYSIS:FFT1:MEASURE:MARKER:

CALCULATION:ALL ON

**:ANALysis:FFT<x>:MEASure:MARKer:****CALCulation:DEFine<x>**

Function Sets the equation of the calculation items of the marker cursor of the FFT computation or queries the current setting.

Syntax :ANALysis:FFT<x>:MEASure:MARKer:

CALCulation:DEFine<x> {<String>}

:ANALysis:FFT<x>:MEASure:MARKer:

CALCulation:DEFine<x>?

<x> of FFT<x> = 1 or 2

<x> of DEFine<x> = 1 to 4

<String> = Up to 128 characters

Example :ANALYSIS:FFT1:MEASURE:MARKER:

CALCULATION:DEFINE1 "V(F1)"

:ANALYSIS:FFT1:MEASURE:MARKER:

CALCULATION:DEFINE1? -> :ANALYSIS:FFT1:

MEASURE:MARKER:CALCULATION:

DEFINE1 "V(F1)"

**:ANALysis:FFT<x>:MEASure:MARKer:****CALCulation:STATe<x>**

Function Turns ON/OFF the calculation items of the marker cursor of the FFT computation or queries the current setting.

Syntax :ANALysis:FFT<x>:MEASure:MARKer:C

ALCulation:STATe<x> {<Boolean>}

:ANALysis:FFT<x>:MEASure:MARKer:

CALCulation:STATe<x>?

<x> of FFT<x> = 1 or 2

<x> of STATe<x> = 1 to 4

Example :ANALYSIS:FFT1:MEASURE:MARKER:

CALCULATION:STATE1 ON

:ANALYSIS:FFT1:MEASURE:MARKER:

CALCULATION:STATE1? -> :ANALYSIS:FFT1:

MEASURE:MARKER:CALCULATION:STATE1 1

### 5.3 ANALYSIS Group

#### **:ANALYSIS:FFT<x>:MEASURE:MARKER:CALCULATION:VALUE<x>?**

**Function** Queries the measured value of the calculation items of the marker cursor of the FFT computation.

**Syntax** :ANALYSIS:FFT<x>:MEASURE:MARKER:CALCULATION:VALUE<x>?

<x> of FFT<x> = 1 or 2

<x> of VALUE<x> = 1 to 4

**Example** :ANALYSIS:FFT1:MEASURE:MARKER:CALCULATION:VALUE1? -> :ANALYSIS:FFT1:MEASURE:MARKER:CALCULATION:VALUE1 1.000E+00

#### **:ANALYSIS:FFT<x>:MEASURE:MODE**

**Function** Sets the automated measurement mode of the FFT computation or queries the current setting.

**Syntax** :ANALYSIS:FFT<x>:MEASURE:MODE {MARKER|OFF|PEAK}

:ANALYSIS:FFT<x>:MEASURE:MODE?

<x> = 1 or 2

**Example** :ANALYSIS:FFT1:MEASURE:MODE MARKER  
:ANALYSIS:FFT1:MEASURE:MODE?  
-> :ANALYSIS:FFT1:MEASURE:MODE MARKER

#### **:ANALYSIS:FFT<x>:MEASURE:PEAK?**

**Function** Queries all settings related to the peak value measurement of the FFT computation.

**Syntax** :ANALYSIS:FFT<x>:MEASURE:PEAK?  
<x> = 1 or 2

**Example** :ANALYSIS:FFT1:MEASURE:PEAK?  
-> :ANALYSIS:FFT1:MEASURE:PEAK:BASIC:DFREQUENCY:STATE 1;:ANALYSIS:FFT1:MEASURE:PEAK:BASIC:DV:STATE 1;:ANALYSIS:FFT1:MEASURE:PEAK:BASIC:FREQUENCY1:STATE 1;:ANALYSIS:FFT1:MEASURE:PEAK:BASIC:FREQUENCY2:STATE 1;:ANALYSIS:FFT1:MEASURE:PEAK:BASIC:RANGE1 1.000E+00,0.000E+00;RANGE2 1.000E+00,0.000E+00;V1:STATE 1;:ANALYSIS:FFT1:MEASURE:PEAK:BASIC:STATE 1;:ANALYSIS:FFT1:MEASURE:PEAK:CALCULATION:DEFINE1 "V(P1)";DEFINE2 "V(P2)";DEFINE3 "V(P1)";DEFINE4 "V(P2)";STATE1 1;STATE2 1;STATE3 1;STATE4 1

#### **:ANALYSIS:FFT<x>:MEASURE:PEAK:BASIC?**

**Function** Queries all settings related to basic items of the peak value of the FFT computation.

**Syntax** :ANALYSIS:FFT<x>:MEASURE:PEAK:BASIC?  
<x> = 1 or 2

**Example** :ANALYSIS:FFT1:MEASURE:PEAK:BASIC?  
-> :ANALYSIS:FFT1:MEASURE:PEAK:BASIC:DFREQUENCY:STATE 1;:ANALYSIS:FFT1:MEASURE:PEAK:BASIC:DV:STATE 1;:ANALYSIS:FFT1:MEASURE:PEAK:BASIC:FREQUENCY1:STATE 1;:ANALYSIS:FFT1:MEASURE:PEAK:BASIC:FREQUENCY2:STATE 1;:ANALYSIS:FFT1:MEASURE:PEAK:BASIC:RANGE1 1.000E+00,0.000E+00;RANGE2 1.000E+00,0.000E+00;V1:STATE 1;:ANALYSIS:FFT1:MEASURE:PEAK:BASIC:V2:STATE 1

#### **:ANALYSIS:FFT<x>:MEASURE:PEAK[:BASIC]:ALL**

**Function** Turns ON/OFF all basic items of the peak value of the FFT computation.

**Syntax** :ANALYSIS:FFT<x>:MEASURE:PEAK[:BASIC]:ALL {<Boolean>}  
<x> = 1 or 2

**Example** :ANALYSIS:FFT1:MEASURE:PEAK:BASIC:ALL ON

#### **:ANALYSIS:FFT<x>:MEASURE:PEAK[:BASIC]:DFREQUENCY?**

**Function** Queries all settings related to the frequency value between peak values of the FFT computation.

**Syntax** :ANALYSIS:FFT<x>:MEASURE:PEAK[:BASIC]:DFREQUENCY?  
<x> = 1 or 2

**Example** :ANALYSIS:FFT1:MEASURE:PEAK:BASIC:DFREQUENCY? -> :ANALYSIS:FFT1:MEASURE:PEAK:BASIC:DFREQUENCY:STATE 1

**:ANALysis:FFT<x>:MEASure:PEAK[:BASic]:DFRequency:STATe**

**Function** Turns ON/OFF the frequency value between peak values of the FFT computation or queries the current setting.

**Syntax** :ANALysis:FFT<x>:MEASure:PEAK[:BASic]:DFRequency:STATe {<Boolean>}  
:ANALysis:FFT<x>:MEASure:PEAK[:BASic]:DFRequency:STATe?  
<x> = 1 or 2

**Example** :ANALYSIS:FFT1:MEASURE:PEAK:BASIC:DFREQUENCY:STATE ON  
:ANALYSIS:FFT1:MEASURE:PEAK:BASIC:DFREQUENCY:STATE? -> :ANALYSIS:FFT1:MEASURE:PEAK:BASIC:DFREQUENCY:STATE 1

**:ANALysis:FFT<x>:MEASure:PEAK[:BASic]:DFRequency:VALue?**

**Function** Queries the frequency value between peak values of the FFT computation.

**Syntax** :ANALysis:FFT<x>:MEASure:PEAK[:BASic]:DFRequency:VALue?  
<x> = 1 or 2

**Example** :ANALYSIS:FFT1:MEASURE:PEAK:BASIC:DFREQUENCY:VALUE? -> :ANALYSIS:FFT1:MEASURE:PEAK:BASIC:DFREQUENCY:VALUE 1.000E+00

**:ANALysis:FFT<x>:MEASure:PEAK[:BASic]:DV?**

**Function** Queries all settings related to the power value between peak values of the FFT computation.

**Syntax** :ANALysis:FFT<x>:MEASure:PEAK[:BASic]:DV?  
<x> = 1 or 2

**Example** :ANALYSIS:FFT1:MEASURE:PEAK:BASIC:DV? -> :ANALYSIS:FFT1:MEASURE:PEAK:BASIC:DV:STATE 1

**:ANALysis:FFT<x>:MEASure:PEAK[:BASic]:DV:STATe**

**Function** Turns ON/OFF the power value between peak values of the FFT computation or queries the current setting.

**Syntax** :ANALysis:FFT<x>:MEASure:PEAK[:BASic]:DV:STATe {<Boolean>}  
:ANALysis:FFT<x>:MEASure:PEAK[:BASic]:DV:STATe?  
<x> = 1 or 2

**Example** :ANALYSIS:FFT1:MEASURE:PEAK:BASIC:DV:STATE ON  
:ANALYSIS:FFT1:MEASURE:PEAK:BASIC:DV:STATE? -> :ANALYSIS:FFT1:MEASURE:PEAK:BASIC:DV:STATE 1

**:ANALysis:FFT<x>:MEASure:PEAK[:BASic]:DV:VALue?**

**Function** Queries the power value between peak values of the FFT computation.

**Syntax** :ANALysis:FFT<x>:MEASure:PEAK[:BASic]:DV:VALue?  
<x> = 1 or 2

**Example** :ANALYSIS:FFT1:MEASURE:PEAK:BASIC:DV:VALUE? -> :ANALYSIS:FFT1:MEASURE:PEAK:BASIC:DV:VALUE 1.000E+00

**:ANALysis:FFT<x>:MEASure:PEAK[:BASic]:FREquency<x>?**

**Function** Queries all settings related to the peak frequency value of the FFT computation.

**Syntax** :ANALysis:FFT<x>:MEASure:PEAK[:BASic]:FREquency<x>?  
<x> of FFT<x> = 1 or 2  
<x> of FREQUENCY<x> = 1 or 2

**Example** :ANALYSIS:FFT1:MEASURE:PEAK:BASIC:FREQUENCY1? -> :ANALYSIS:FFT1:MEASURE:PEAK:BASIC:FREQUENCY1:STATE 1

**:ANALysis:FFT<x>:MEASure:PEAK[:BASic]:FREquency<x>:STATe**

**Function** Turns ON/OFF the peak frequency value of the FFT computation or queries the current setting.

**Syntax** :ANALysis:FFT<x>:MEASure:PEAK[:BASic]:FREquency<x>:STATe {<Boolean>}  
:ANALysis:FFT<x>:MEASure:PEAK[:BASic]:FREquency<x>:STATe?  
<x> of FFT<x> = 1 or 2  
<x> of FREQUENCY<x> = 1 or 2

**Example** :ANALYSIS:FFT1:MEASURE:PEAK:BASIC:FREQUENCY1:STATE ON  
:ANALYSIS:FFT1:MEASURE:PEAK:BASIC:FREQUENCY1:STATE? -> :ANALYSIS:FFT1:MEASURE:PEAK:BASIC:FREQUENCY1:STATE 1

**:ANALysis:FFT<x>:MEASure:PEAK[:BASic]:FREquency<x>:VALue?**

**Function** Queries the peak frequency value of the FFT computation.

**Syntax** :ANALysis:FFT<x>:MEASure:PEAK[:BASic]:FREquency<x>:VALue?  
<x> of FFT<x> = 1 or 2  
<x> of FREQUENCY<x> = 1 or 2

**Example** :ANALYSIS:FFT1:MEASURE:PEAK:BASIC:FREQUENCY1:VALUE? -> :ANALYSIS:FFT1:MEASURE:PEAK:BASIC:FREQUENCY1:VALUE 1.000E+00

### 5.3 ANALysis Group

#### **:ANALysis:FFT<x>:MEASure:PEAK[:BASic]:RANGe<x>**

**Function** Sets the measurement range of the peak value of the FFT computation or queries the current setting.

**Syntax** :ANALysis:FFT<x>:MEASure:PEAK[:BASic]:RANGe<x> {<NRf>,<NRf>}  
:ANALysis:FFT<x>:MEASure:PEAK[:BASic]:RANGe<x>?  
<x> of FFT<x> = 1 or 2  
<x> of RANGe<x> = 1 or 2  
<NRf> = -5 to 5 div

**Example** :ANALYSIS:FFT1:MEASURE:PEAK:BASIC:  
RANGE1 0,1  
:ANALYSIS:FFT1:MEASURE:PEAK:BASIC:  
RANGE1? -> :ANALYSIS:FFT1:MEASURE:  
PEAK:BASIC:RANGE1 1.000E+00,0.000E+00

#### **:ANALysis:FFT<x>:MEASure:PEAK[:BASic]:V<x>?**

**Function** Queries all settings related to the peak value of the FFT computation.

**Syntax** :ANALysis:FFT<x>:MEASure:PEAK[:BASic]:V<x>?  
<x> of FFT<x> = 1 or 2  
<x> of V<x> = 1 or 2

**Example** :ANALYSIS:FFT1:MEASURE:PEAK:BASIC:V1?  
-> :ANALYSIS:FFT1:MEASURE:PEAK:BASIC:  
V1:STATE 1

#### **:ANALysis:FFT<x>:MEASure:PEAK[:BASic]:V<x>:STATE**

**Function** Turns ON/OFF the peak value of the FFT computation or queries the current setting.

**Syntax** :ANALysis:FFT<x>:MEASure:PEAK[:BASic]:V<x>:STATE {<Boolean>}  
:ANALysis:FFT<x>:MEASure:PEAK[:BASic]:V<x>:STATE?  
<x> of FFT<x> = 1 or 2  
<x> of V<x> = 1 or 2

**Example** :ANALYSIS:FFT1:MEASURE:PEAK:BASIC:V1:  
STATE ON  
:ANALYSIS:FFT1:MEASURE:PEAK:BASIC:V1:  
STATE? -> :ANALYSIS:FFT1:MEASURE:PEAK:  
BASIC:V1:STATE 1

#### **:ANALysis:FFT<x>:MEASure:PEAK[:BASic]:V<x>:VALue?**

**Function** Queries the peak value of the FFT computation.

**Syntax** :ANALysis:FFT<x>:MEASure:PEAK[:BASic]:V<x>:VALue?  
<x> of FFT<x> = 1 or 2  
<x> of V<x> = 1 or 2

**Example** :ANALYSIS:FFT1:MEASURE:PEAK:BASIC:V1:  
VALUE? -> :ANALYSIS:FFT1:MEASURE:PEAK:  
BASIC:V1:VALUE 1.000E+00

#### **:ANALysis:FFT<x>:MEASure:PEAK:CALCulation?**

**Function** Queries all settings related to calculation items of the FFT computation.

**Syntax** :ANALysis:FFT<x>:MEASure:PEAK:CALCulation?  
<x> = 1 or 2

**Example** :ANALYSIS:FFT1:MEASURE:PEAK:  
CALCULATION? -> :ANALYSIS:FFT1:MEASURE:  
PEAK:CALCULATION:DEFINE1 "V(P1)";  
DEFINE2 "V(P2)";DEFINE3 "V(P1)";  
DEFINE4 "V(P2)";STATE1 1;STATE2 1;  
STATE3 1;STATE4 1

#### **:ANALysis:FFT<x>:MEASure:PEAK:CALCulation:ALL**

**Function** Turns ON/OFF all calculation items of the FFT computation.

**Syntax** :ANALysis:FFT<x>:MEASure:PEAK:CALCulation:ALL {<Boolean>}  
<x> = 1 or 2

**Example** :ANALYSIS:FFT1:MEASURE:PEAK:  
CALCULATION:ALL ON

#### **:ANALysis:FFT<x>:MEASure:PEAK:CALCulation:DEFine<x>**

**Function** Sets the equation of the calculation item of the FFT computation or queries the current setting.

**Syntax** :ANALysis:FFT<x>:MEASure:PEAK:CALCulation:DEFine<x> {<String>}  
:ANALysis:FFT<x>:MEASure:PEAK:CALCulation:DEFine<x>?  
<x> of FFT<x> = 1 or 2  
<x> of DEFine<x> = 1 to 4  
<String> = Up to 128 characters

**Example** :ANALYSIS:FFT1:MEASURE:PEAK:  
CALCULATION:DEFINE1 "V(P1)"  
:ANALYSIS:FFT1:MEASURE:PEAK:  
CALCULATION:DEFINE1? -> :ANALYSIS:  
FFT1:MEASURE:PEAK:CALCULATION:DEFINE1  
"V(P1)"

**:ANALYSIS:FFT<x>:MEASURE:PEAK:  
CALCULATION:STATE<x>**

**Function** Turns ON/OFF the calculation items of the FFT computation or queries the current setting.

**Syntax** :ANALYSIS:FFT<x>:MEASURE:PEAK:  
CALCULATION:STATE<x> {<Boolean>}  
:ANALYSIS:FFT<x>:MEASURE:PEAK:  
CALCULATION:STATE<x>?  
<x> of FFT<x> = 1 or 2  
<x> of STATE<x> = 1 to 4

**Example** :ANALYSIS:FFT1:MEASURE:PEAK:  
CALCULATION:STATE1 ON  
:ANALYSIS:FFT1:MEASURE:PEAK:  
CALCULATION:STATE1? -> :ANALYSIS:FFT1:  
MEASURE:PEAK:CALCULATION:STATE1 1

**:ANALYSIS:FFT<x>:MEASURE:PEAK:  
CALCULATION:VALUE<x>?**

**Function** Queries the measured value of the calculation item of the FFT computation.

**Syntax** :ANALYSIS:FFT<x>:MEASURE:PEAK:  
CALCULATION:VALUE<x>?  
<x> of FFT<x> = 1 or 2  
<x> of VALUE<x> = 1 to 4

**Example** :ANALYSIS:FFT1:MEASURE:PEAK:  
CALCULATION:VALUE1? -> :ANALYSIS:FFT1:  
MEASURE:PEAK:CALCULATION:  
VALUE1 1.000E+00

**:ANALYSIS:FFT<x>:RANGE**

**Function** Sets the measurement source window used in the FFT computation or queries the current setting.

**Syntax** :ANALYSIS:FFT<x>:RANGE {MAIN|Z1|Z2}  
:ANALYSIS:FFT<x>:RANGE?  
<x> = 1 or 2

**Example** :ANALYSIS:FFT1:RANGE MAIN  
:ANALYSIS:FFT1:RANGE? -> :ANALYSIS:  
FFT1:RANGE MAIN

**:ANALYSIS:FFT<x>:RPART (Real Part)**

**Function** Sets the source trace of the real part of the FFT computation or queries the current setting.

**Syntax** :ANALYSIS:FFT<x>:RPART {<NRf>}  
:ANALYSIS:FFT<x>:RPART?  
<x> = 1 or 2  
<NRf> = 1 to 8

**Example** :ANALYSIS:FFT1:RPART 1  
:ANALYSIS:FFT1:RPART? -> :ANALYSIS:  
FFT1:RPART 1

**:ANALYSIS:FFT<x>:RPOSITION (Ref  
Position)**

**Function** Sets the center point of magnification of the vertical axis of the FFT computation or queries the current setting.

**Syntax** :ANALYSIS:FFT<x>:RPOSITION {<NRf>}  
:ANALYSIS:FFT<x>:RPOSITION?  
<x> = 1 or 2  
<NRf> = -4 to 4 (div)

**Example** :ANALYSIS:FFT1:RPOSITION 1  
:ANALYSIS:FFT1:RPOSITION? -> :ANALYSIS:  
FFT1:RPOSITION 1.000E+00

**:ANALYSIS:FFT<x>:VERTICAL?**

**Function** Queries all settings related the vertical axis of the FFT computation.

**Syntax** :ANALYSIS:FFT<x>:VERTICAL?  
<x> = 1 or 2

**Example** :ANALYSIS:FFT1:VERTICAL? -> :ANALYSIS:  
FFT1:VERTICAL:  
LEVEL 1.000E+00;MODE AUTO;  
SENSITIVITY 1.000E+00

**:ANALYSIS:FFT<x>:VERTICAL:LEVEL**

**Function** Sets the display position of the vertical axis of the FFT computation or queries the current setting.

**Syntax** :ANALYSIS:FFT<x>:VERTICAL:LEVEL {<NRf>}  
:ANALYSIS:FFT<x>:VERTICAL:LEVEL?  
<x> = 1 or 2  
<NRf> = -1.000E+31 to 1.000E+31 (dBV)

**Example** :ANALYSIS:FFT1:VERTICAL:LEVEL 1  
:ANALYSIS:FFT1:VERTICAL:LEVEL?  
-> :ANALYSIS:FFT1:VERTICAL:  
LEVEL 1.000E+00

**:ANALYSIS:FFT<x>:VERTICAL:MODE**

**Function** Sets the vertical axis mode of the FFT computation or queries the current setting.

**Syntax** :ANALYSIS:FFT<x>:VERTICAL:MODE {AUTO|  
MANUAL}  
:ANALYSIS:FFT<x>:VERTICAL:MODE?  
<x> = 1 or 2

**Example** :ANALYSIS:FFT1:VERTICAL:MODE AUTO  
:ANALYSIS:FFT1:VERTICAL:MODE?  
-> :ANALYSIS:FFT1:VERTICAL:MODE AUTO

### 5.3 ANALysis Group

#### **:ANALysis:FFT<x>:VERTical:SENSitivity**

**Function** Sets the vertical sensitivity of the FFT computation or queries the current setting.

**Syntax** :ANALysis:FFT<x>:VERTical:SENSitivity {<NRF>}  
:ANALysis:FFT<x>:VERTical:SENSitivity?<x> = 1 or 2<NRF> = 0 to 1.000E+31 (dBV)

**Example** :ANALYSIS:FFT1:VERTICAL:SENSITIVITY 1  
:ANALYSIS:FFT1:VERTICAL:SENSITIVITY?  
-> :ANALYSIS:FFT1:VERTICAL:SENSITIVITY 1.000E+00

#### **:ANALysis:FFT<x>:WINDow**

**Function** Sets the window function of the FFT computation or queries the current setting.

**Syntax** :ANALysis:FFT<x>:WINDow {FLATtop|HANNing|RECTangle}  
:ANALysis:FFT<x>:WINDow?<x> = 1 or 2

**Example** :ANALYSIS:FFT1:WINDOW FLATTOP  
:ANALYSIS:FFT1:WINDOW? -> :ANALYSIS:FFT1:WINDOW FLATTOP

#### **:ANALysis:TYPE<x>**

**Function** Sets the analysis function type or queries the current setting.

**Syntax** :ANALysis:TYPE<x> {AHISTogram|FFT|WPARAMeter|XY}  
:ANALysis:TYPE<x>?<x> = 1 or 2

**Example** :ANALYSIS:TYPE1 AHISTOGRAM  
:ANALYSIS:TYPE1? -> :ANALYSIS:TYPE1 AHISTOGRAM

#### **:ANALysis:VTDisPlay**

**Function** Turns ON/OFF the VT waveform display or queries the current setting.

**Syntax** :ANALysis:VTDisPlay {<Boolean>}  
:ANALysis:VTDisPlay?

**Example** :ANALYSIS:VTDISPLAY ON  
:ANALYSIS:VTDISPLAY? -> :ANALYSIS:VTDISPLAY 1

#### **:ANALysis:WAIT<x>?**

**Function** Waits for the completion of the automated measurement with a timeout option.

**Syntax** ANALysis:WAIT<x>? {<NRF>}<x> = 1 or 2<NRF> = 1 to 360000 (timeout value, in units of 10 ms)

**Example** ANALYSIS:WAIT1? 100 -> :ANALYSIS:WAIT1 1

**Description** If the execution of the automated measurement completes within the timeout value, 0 is returned; if it is not complete or automated measurement is not being executed, 1 is returned. Even if the timeout value is set long, 0 is returned when the execution of the automated measurement is complete.

#### **:ANALysis:WPARAMeter<x>?**

**Function** Queries all settings related to the waveform parameter measurement function.

**Syntax** :ANALysis:WPARAMeter<x>?<x> = 1 or 2

**Example** :ANALYSIS:WPARAMETER1? -> :ANALYSIS:WPARAMETER1:CALCULATION 1;HISTOGRAM:MEASURE:MODE CURSOR;CURSOR:C1:POSITION 1.000E+00;STATE 1;:ANALYSIS:WPARAMETER1:HISTOGRAM:MEASURE:CURSOR:C2:POSITION 2.000E+00;STATE 1;:ANALYSIS:WPARAMETER1:HISTOGRAM:MEASURE:CURSOR:DC:STATE 1;:ANALYSIS:WPARAMETER1:HISTOGRAM:MEASURE:CURSOR:LINKAGE 1;:ANALYSIS:WPARAMETER1:HISTOGRAM:MEASURE:PARAMETER:MEAN:STATE 1;:ANALYSIS:WPARAMETER1:HISTOGRAM:MEASURE:PARAMETER:PEAK:STATE 1;:ANALYSIS:WPARAMETER1:HISTOGRAM:MEASURE:PARAMETER:SD3INTEG:STATE 1;:ANALYSIS:WPARAMETER1:HISTOGRAM:MEASURE:PARAMETER:SDEVIATION:STATE 1;:ANALYSIS:WPARAMETER1:HISTOGRAM:MEASURE:PARAMETER:SDINTEG:STATE 1;:ANALYSIS:WPARAMETER1:LIST:MODE STATISTIC;SCROLL HORIZONTAL;:ANALYSIS:WPARAMETER1:MODE HISTOGRAM;TREND:HRANGE MAIN;HSPAN 1;MEASURE:CURSOR:C1:POSITION 1.000E+00;STATE 1;:ANALYSIS:WPARAMETER1:TREND:MEASURE:CURSOR:C2:POSITION 2.000E+00;STATE 1;:ANALYSIS:WPARAMETER1:TREND:MEASURE:CURSOR:DC:STATE 1;:ANALYSIS:WPARAMETER1:TREND:MEASURE:MODE CURSOR;:ANALYSIS:WPARAMETER1:TREND:VERTICAL 2.000E+00,1.000E+00

**:ANALysis:WPARAMeter<x>:CALCulation**

**Function** Sets the calculation items of the automated measurement of waveform parameters or queries the current setting.

**Syntax** :ANALysis:WPARAMeter<x>:CALCulation  
{<NRf>}  
:ANALysis:WPARAMeter<x>:CALCulation?  
<x> = 1 or 2  
<NRf> = 1 to 4

**Example** :ANALYSIS:WPARAMETER1:CALCULATION 1  
:ANALYSIS:WPARAMETER1:CALCULATION?  
-> :ANALYSIS:WPARAMETER1:CALCULATION 1

**:ANALysis:WPARAMeter<x>:HISTogram?**

**Function** Queries all settings related to the histogram display of the waveform parameter measurement.

**Syntax** :ANALysis:WPARAMeter<x>:HISTogram?  
<x> = 1 or 2

**Example** :ANALYSIS:WPARAMETER1:HISTOGRAM?  
-> :ANALYSIS:WPARAMETER1:HISTOGRAM:  
MEASURE:MODE CURSOR;CURSOR:C1:  
POSITION 1.000E+00;STATE 1;  
:ANALYSIS:WPARAMETER1:HISTOGRAM:  
MEASURE:CURSOR:C2:  
POSITION 2.000E+00;STATE 1;  
:ANALYSIS:WPARAMETER1:HISTOGRAM:  
MEASURE:CURSOR:DC:STATE 1;  
:ANALYSIS:WPARAMETER1:HISTOGRAM:  
MEASURE:CURSOR:LINKAGE 1;  
:ANALYSIS:WPARAMETER1:HISTOGRAM:  
MEASURE:PARAMETER:MEAN:STATE 1;  
:ANALYSIS:WPARAMETER1:HISTOGRAM:  
MEASURE:PARAMETER:PEAK:STATE 1;  
:ANALYSIS:WPARAMETER1:HISTOGRAM:  
MEASURE:PARAMETER:SD3INTEG:STATE 1;  
:ANALYSIS:WPARAMETER1:HISTOGRAM:  
MEASURE:PARAMETER:SDEVIATION:STATE 1;  
:ANALYSIS:WPARAMETER1:HISTOGRAM:  
MEASURE:PARAMETER:SDINTEG:STATE 1

**:ANALysis:WPARAMeter<x>:HISTogram:MEASure?**

**Function** Queries all settings related to the automated measurement of the histogram display of the waveform parameter measurement.

**Syntax** :ANALysis:WPARAMeter<x>:HISTogram:  
MEASure?  
<x> = 1 or 2

**Example** :ANALYSIS:WPARAMETER1:HISTOGRAM:  
MEASURE? -> :ANALYSIS:WPARAMETER1:  
HISTOGRAM:  
MEASURE:MODE CURSOR;CURSOR:C1:  
POSITION 1.000E+00;STATE 1;  
:ANALYSIS:WPARAMETER1:HISTOGRAM:  
MEASURE:CURSOR:C2:POSITION 2.000E+00;  
STATE 1;  
:ANALYSIS:WPARAMETER1:HISTOGRAM:  
MEASURE:CURSOR:DC:STATE 1;  
:ANALYSIS:WPARAMETER1:HISTOGRAM:  
MEASURE:CURSOR:LINKAGE 1;  
:ANALYSIS:WPARAMETER1:HISTOGRAM:  
MEASURE:PARAMETER:MEAN:STATE 1;  
:ANALYSIS:WPARAMETER1:HISTOGRAM:  
MEASURE:PARAMETER:PEAK:STATE 1;  
:ANALYSIS:WPARAMETER1:HISTOGRAM:  
MEASURE:PARAMETER:SD3INTEG:STATE 1;  
:ANALYSIS:WPARAMETER1:HISTOGRAM:  
MEASURE:PARAMETER:SDEVIATION:STATE 1;  
:ANALYSIS:WPARAMETER1:HISTOGRAM:  
MEASURE:PARAMETER:SDINTEG:STATE 1

**:ANALysis:WPARAMeter<x>:HISTogram:MEASure:CURSor?**

**Function** Queries all settings related to the cursor measurement on the histogram of the waveform parameter measurement.

**Syntax** :ANALysis:WPARAMeter<x>:HISTogram:  
MEASure:CURSor?  
<x> = 1 or 2

**Example** :ANALYSIS:WPARAMETER1:HISTOGRAM:  
MEASURE:CURSOR? -> :ANALYSIS:  
WPARAMETER1:HISTOGRAM:  
MEASURE:CURSOR:C1:POSITION 1.000E+00;  
STATE 1;  
:ANALYSIS:WPARAMETER1:HISTOGRAM:  
MEASURE:CURSOR:C2:POSITION 2.000E+00;  
STATE 1;  
:ANALYSIS:WPARAMETER1:HISTOGRAM:  
MEASURE:CURSOR:DC:STATE 1;  
:ANALYSIS:WPARAMETER1:HISTOGRAM:  
MEASURE:CURSOR:LINKAGE 1

### 5.3 ANALysis Group

#### **:ANALysis:WPARAMeter<x>:HISTogram:**

##### **MEASure:CURSor:ALL**

**Function** Turns ON/OFF all histogram cursors of waveform parameter measurement.

**Syntax** :ANALysis:WPARAMeter<x>:HISTogram:  
MEASure:CURSor:ALL {<Boolean>}  
<x> = 1 or 2

**Example** :ANALYSIS:WPARAMETER1:HISTOGRAM:  
MEASURE:CURSOR:ALL ON

#### **:ANALysis:WPARAMeter<x>:HISTogram:**

##### **MEASure:CURSor:C<x>?**

**Function** Queries all settings related to the cursor measurement on the histogram of the waveform parameter measurement.

**Syntax** :ANALysis:WPARAMeter<x>:HISTogram:  
MEASure:CURSor:C<x>?  
<x> of WPARAMeter<x> = 1 or 2  
<x> of C<x> = 1 or 2

**Example** :ANALYSIS:WPARAMETER1:HISTOGRAM:  
MEASURE:CURSOR:C1? -> :ANALYSIS:  
WPARAMETER1:HISTOGRAM:  
MEASURE:CURSOR:C1:POSITION 1.000E+00;  
STATE 1

#### **:ANALysis:WPARAMeter<x>:HISTogram:**

##### **MEASure:CURSor:C<x>:POSITION**

**Function** Sets the cursor position on the trend of the waveform parameter measurement or queries the current setting.

**Syntax** :ANALysis:WPARAMeter<x>:HISTogram:  
MEASure:PARAMeter:C<x>:POSITION {<NRf>}  
:ANALysis:WPARAMeter<x>:HISTogram:  
MEASure:PARAMeter:C<x>:POSITION?  
<x> of WPARAMeter<x> = 1 or 2  
<x> of C<x> = 1 or 2  
<NRf> = -5 to 5 (div)

**Example** :ANALYSIS:WPARAMETER1:HISTOGRAM:  
MEASURE:CURSOR:C1:POSITION ON  
:ANALYSIS:WPARAMETER1:HISTOGRAM:  
MEASURE:CURSOR:C1:POSITION?  
-> :ANALYSIS:WPARAMETER1:HISTOGRAM:  
MEASURE:CURSOR:C1:POSITION 1.000E+00

#### **:ANALysis:WPARAMeter<x>:HISTogram:**

##### **MEASure:CURSor:C<x>:STATE**

**Function** Turns ON/OFF the cursor on the histogram of the waveform parameter measurement or queries the current setting.

**Syntax** :ANALysis:WPARAMeter<x>:HISTogram:  
MEASure:PARAMeter:C<x>:  
STATE {<Boolean>}  
:ANALysis:WPARAMeter<x>:HISTogram:  
MEASure:PARAMeter:C<x>:STATE?  
<x> of WPARAMeter<x> = 1 or 2  
<x> of C<x> = 1 or 2

**Example** :ANALYSIS:WPARAMETER1:HISTOGRAM:  
MEASURE:CURSOR:C1:STATE ON  
:ANALYSIS:WPARAMETER1:HISTOGRAM:  
MEASURE:CURSOR:C1:STATE? -> :ANALYSIS:  
WPARAMETER1:HISTOGRAM:MEASURE:CURSOR:  
C1:STATE 1

#### **:ANALysis:WPARAMeter<x>:HISTogram:**

##### **MEASure:CURSor:C<x>:VALUE?**

**Function** Queries the measured value of the cursor on the histogram of the waveform parameter measurement.

**Syntax** :ANALysis:WPARAMeter<x>:HISTogram:  
MEASure:CURSor:C<x>:VALUE?  
<x> of WPARAMeter<x> = 1 or 2  
<x> of C<x> = 1 or 2

**Example** :ANALYSIS:WPARAMETER1:HISTOGRAM:  
MEASURE:CURSOR:C1:VALUE? -> :ANALYSIS:  
WPARAMETER1:HISTOGRAM:MEASURE:CURSOR:  
C1:VALUE 1.000E+00

#### **:ANALysis:WPARAMeter<x>:HISTogram:**

##### **MEASure:CURSor:DC?**

**Function** Queries all settings related to the measurement between cursors on the histogram of the waveform parameter measurement.

**Syntax** :ANALysis:WPARAMeter<x>:HISTogram:  
MEASure:CURSor:DC?  
<x> = 1 or 2

**Example** :ANALYSIS:WPARAMETER1:HISTOGRAM:  
MEASURE:CURSOR:DC? -> :ANALYSIS:  
WPARAMETER1:HISTOGRAM:MEASURE:CURSOR:  
DC:STATE 1



**:ANALysis:WPARAmeter<x>:HISTogram:****MEASure:CURSor:DC:STATE**

**Function** Turns ON/OFF the measurement between cursors on the histogram of the waveform parameter measurement or queries the current setting.

**Syntax** :ANALysis:WPARAmeter<x>:HISTogram:  
MEASure:PARAmeter:DC:STATe {<Boolean>}  
:ANALysis:WPARAmeter<x>:HISTogram:  
MEASure:PARAmeter:DC:STATe?  
<x> = 1 or 2

**Example** :ANALYSIS:WPARAMETER1:HISTOGRAM:  
MEASURE:CURSOR:DC:STATE ON  
:ANALYSIS:WPARAMETER1:HISTOGRAM:  
MEASURE:CURSOR:DC:STATE? -> :ANALYSIS:  
WPARAMETER1:HISTOGRAM:MEASURE:CURSOR:  
DC:STATE 1

**:ANALysis:WPARAmeter<x>:HISTogram:****MEASure:CURSor:DC:VALue?**

**Function** Queries the measured value between cursors on the histogram of the waveform parameter measurement.

**Syntax** :ANALysis:WPARAmeter<x>:HISTogram:  
MEASure:CURSor:DC:VALue?  
<x> = 1 or 2

**Example** :ANALYSIS:WPARAMETER1:HISTOGRAM:  
MEASURE:CURSOR:DC:VALUE? -> :ANALYSIS:  
WPARAMETER1:HISTOGRAM:MEASURE:CURSOR:  
DC:VALUE 1.000E+00

**:ANALysis:WPARAmeter<x>:HISTogram:****MEASure:CURSor:LINKage**

**Function** Turns ON/OFF the histogram cursor link of the waveform parameter measurement or queries the current setting.

**Syntax** :ANALysis:WPARAmeter<x>:HISTogram:  
MEASure:CURSor:LINKage {<Boolean>}  
:ANALysis:WPARAmeter<x>:HISTogram:  
MEASure:CURSor:LINKage?  
<x> = 1 or 2

**Example** :ANALYSIS:WPARAMETER1:HISTOGRAM:  
MEASURE:CURSOR:LINKAGE ON  
:ANALYSIS:WPARAMETER1:HISTOGRAM:  
MEASURE:CURSOR:LINKAGE? -> :ANALYSIS:  
WPARAMETER1:HISTOGRAM:MEASURE:CURSOR:  
LINKAGE 1

**:ANALysis:WPARAmeter<x>:HISTogram:****MEASure:MODE**

**Function** Sets the automated measurement mode of the histogram display of the waveform parameter measurement or queries the current setting.

**Syntax** :ANALysis:WPARAmeter<x>:HISTogram:  
MEASure:MODE {CURSor|OFF|PARAmeter}  
:ANALysis:WPARAmeter<x>:HISTogram:  
MEASure:MODE?  
<x> = 1 or 2

**Example** :ANALYSIS:WPARAMETER1:HISTOGRAM:  
MEASURE:MODE CURSOR  
:ANALYSIS:WPARAMETER1:HISTOGRAM:  
MEASURE:MODE? -> :ANALYSIS:WPARAMETER1:  
HISTOGRAM:MEASURE:MODE CURSOR

**:ANALysis:WPARAmeter<x>:HISTogram:****MEASure:PARAmeter?**

**Function** Queries all settings related to the automated measurement of histogram parameters of the waveform parameter measurement.

**Syntax** :ANALysis:WPARAmeter<x>:HISTogram:  
MEASure:PARAmeter?  
<x> = 1 or 2

**Example** :ANALYSIS:WPARAMETER1:HISTOGRAM:  
MEASURE:PARAMETER? -> :ANALYSIS:  
WPARAMETER1:HISTOGRAM:  
MEASURE:PARAMETER:MEAN:STATE 1;:  
ANALYSIS:WPARAMETER1:HISTOGRAM:MEASURE:  
PARAMETER:PEAK:STATE 1;:ANALYSIS:  
WPARAMETER1:HISTOGRAM:MEASURE:  
PARAMETER:SD3INTEG:STATE 1;:ANALYSIS:  
WPARAMETER1:HISTOGRAM:MEASURE:  
PARAMETER:SDEVIATION:STATE 1;:ANALYSIS:  
WPARAMETER1:HISTOGRAM:MEASURE:  
PARAMETER:SDINTEG:STATE 1

**:ANALysis:WPARAmeter<x>:HISTogram:****MEASure:PARAmeter:ALL**

**Function** Turn ON/OFF all histogram parameters of the waveform parameter measurement.

**Syntax** :ANALysis:WPARAmeter<x>:HISTogram:  
MEASure:PARAmeter:ALL {<Boolean>}  
<x> = 1 or 2

**Example** :ANALYSIS:WPARAMETER1:HISTOGRAM:  
MEASURE:PARAMETER:ALL ON

### 5.3 ANALysis Group

#### **:ANALysis:WPARAMeter<x>:HISTogram:**

##### **MEASure:PARAmeter:<Parameter>?**

**Function** Queries all settings related to the histogram parameter of the waveform parameter measurement.

**Syntax** :ANALysis:WPARAMeter<x>:HISTogram:

MEASure:PARAmeter:<Parameter>?

<x> = 1 or 2

<Parameter> = {MEAN|PEAK|SD3integ|

SDEVIation|SDINteg}

**Example** (The following is an example for the average value.)

```
:ANALYSIS:WPARAMETER1:HISTOGRAM:
MEASURE:PARAMETER:MEAN? -> :ANALYSIS:
WPARAMETER1:HISTOGRAM:MEASURE:
PARAMETER:MEAN:STATE 1
```

#### **:ANALysis:WPARAMeter<x>:HISTogram:**

##### **MEASure:PARAmeter:<Parameter>:STATe**

**Function** Turns ON/OFF the histogram parameter of the waveform parameter measurement or queries the current setting.

**Syntax** :ANALysis:WPARAMeter<x>:HISTogram:

MEASure:PARAmeter:<Parameter>:STATe

{<Boolean>}

:ANALysis:WPARAMeter<x>:HISTogram:

MEASure:PARAmeter:<Parameter>:STATe?

<x> = 1 or 2

<Parameter> = {MEAN|PEAK|SD3integ|

SDEVIation|SDINteg}

**Example** (The following is an example for the average value.)

```
:ANALYSIS:WPARAMETER1:HISTOGRAM:
MEASURE:PARAMETER:MEAN:STATE ON
:ANALYSIS:WPARAMETER1:HISTOGRAM:
MEASURE:PARAMETER:MEAN:STATE?
-> :ANALYSIS:WPARAMETER1:HISTOGRAM:
MEASURE:PARAMETER:MEAN:STATE 1
```

#### **:ANALysis:WPARAMeter<x>:HISTogram:**

##### **MEASure:PARAmeter:<Parameter>:VALue?**

**Function** Queries the measured value of the histogram parameter of the waveform parameter measurement.

**Syntax** :ANALysis:WPARAMeter<x>:HISTogram:

MEASure:PARAmeter:<Parameter>:VALue?

<x> = 1 or 2

<Parameter> = {MEAN|PEAK|SD3integ|

SDEVIation|SDINteg}

**Example** (The following is an example for the average value.)

```
:ANALYSIS:WPARAMETER1:HISTOGRAM:
MEASURE:PARAMETER:MEAN:VALUE?
-> :ANALYSIS:WPARAMETER1:HISTOGRAM:
MEASURE:PARAMETER:MEAN:VALUE 1.000E+00
```

#### **:ANALysis:WPARAMeter<x>:LIST?**

**Function** Queries all settings related to the list display of the waveform parameter measurement.

**Syntax** :ANALysis:WPARAMeter<x>:LIST?

<x> = 1 or 2

**Example** :ANALYSIS:WPARAMETER1:LIST?

```
-> :ANALYSIS:WPARAMETER1:LIST:
MODE STATISTIC;SCROLL HORIZONTAL
```

#### **:ANALysis:WPARAMeter<x>:LIST:ITEM?**

**Function** Queries list display items of the waveform parameter measurement.

**Syntax** :ANALysis:WPARAMeter<x>:LIST:ITEM?

<x> = 1 or 2

**Example** :ANALYSIS:WPARAMETER1:LIST:ITEM?

```
-> :ANALYSIS:WPARAMETER1:LIST:
ITEM "LOW(C1)"
```

#### **:ANALysis:WPARAMeter<x>:LIST:MODE**

**Function** Sets the list display mode of the waveform parameter measurement or queries the current setting.

**Syntax** :ANALysis:WPARAMeter<x>:LIST:MODE

{STATistic|TREND}

:ANALysis:WPARAMeter<x>:LIST:MODE?

<x>=1 or 2

**Example** :ANALYSIS:WPARAMETER1:LIST:MODE

```
STATISTIC
:ANALYSIS:WPARAMETER1:LIST:MODE?
-> :ANALYSIS:WPARAMETER1:LIST:
MODE STATISTIC
```

#### **:ANALysis:WPARAMeter<x>:LIST:SCRoll**

**Function** Sets the scroll direction of the list display of the waveform parameter measurement or queries the current setting.

**Syntax** :ANALysis:WPARAMeter<x>:LIST:SCRoll

{HORizontal|VERTical}

:ANALysis:WPARAMeter<x>:LIST:SCRoll?

<x> = 1 or 2

**Example** :ANALYSIS:WPARAMETER1:LIST:

```
SCROLL HORIZONTAL
:ANALYSIS:WPARAMETER1:LIST:SCROLL?
-> :ANALYSIS:WPARAMETER1:LIST:
SCROLL HORIZONTAL
```

**:ANALYSIS:WPARAMETER<x>:LIST:VALUE?**

**Function** Queries the automated measured value of the list display number of the waveform parameter measurement.

**Syntax** :ANALYSIS:WPARAMETER<x>:LIST:VALUE? {<NRF>|MAXIMUM}  
<x> = 1 or 2  
<NRF> = 1 to 100000

**Example** :ANALYSIS:WPARAMETER1:LIST:VALUE? 1  
-> :ANALYSIS:WPARAMETER1:LIST:VALUE 1,1.000E+00

**Description** • If the measurement is not possible, "NAN (Not A Number)" is returned.

- If the value of the list display number is not present, "NAN (Not A Number)" is returned.
- If MAXIMUM is specified, the maximum list display number is selected.

**:ANALYSIS:WPARAMETER<x>:MODE**

**Function** Sets the mode of the waveform parameter measurement or queries the current setting.

**Syntax** :ANALYSIS:WPARAMETER<x>:MODE {HISTOGRAM|LIST|TRENd}  
:ANALYSIS:WPARAMETER<x>:MODE?  
<x> = 1 or 2

**Example** :ANALYSIS:WPARAMETER1:MODE HISTOGRAM  
:ANALYSIS:WPARAMETER1:MODE?  
-> :ANALYSIS:WPARAMETER1:MODE HISTOGRAM

**:ANALYSIS:WPARAMETER<x>:TRACE<x>?**

**Function** Queries all settings related to the trace of the waveform parameter measurement.

**Syntax** :ANALYSIS:WPARAMETER<x>:TRACE<x>?  
<x> of WPARAMETER<x> = 1 or 2  
<x> of TRACE<x> = 1 to 8

**Example** :ANALYSIS:WPARAMETER1:TRACE1?  
-> :ANALYSIS:WPARAMETER1:TRACE1:AREA1:TYPE BURST

**:ANALYSIS:WPARAMETER<x>:TRACE<x>:AREA<x>?**

**Function** Queries all settings related to the area of the waveform parameter measurement.

**Syntax** :ANALYSIS:WPARAMETER<x>:TRACE<x>:AREA<x>?  
<x> of WPARAMETER<x> = 1 or 2  
<x> of TRACE<x> = 1 to 8  
<x> of AREA<x> = 1 or 2

**Example** :ANALYSIS:WPARAMETER1:TRACE1:AREA1?  
-> :ANALYSIS:WPARAMETER1:TRACE1:AREA1:TYPE BURST

**:ANALYSIS:WPARAMETER<x>:TRACE<x>:****AREA<x>:TYPE**

**Function** Sets the waveform parameter of the waveform parameter measurement or queries the current setting.

**Syntax** :ANALYSIS:WPARAMETER<x>:TRACE<x>:AREA<x>:TYPE {<Parameter>}  
:ANALYSIS:WPARAMETER<x>:TRACE<x>:AREA<x>:TYPE?

<x> of WPARAMETER<x> = 1 or 2

<x> of TRACE<x> = 1 to 8

<x> of AREA<x> = 1 or 2

<Parameter> = {BURSt|CMEan|COUNT|CRMS|CSDeViation|DElay|DT|DUTYcycle|FALL|FREQuency|HIGH|HILow|LOW|MAXimum|MEAN|MINimum|NOVershoot|NWIDTH|PERFrequency|PERiod|POVershoot|PTOPeak|PWIDth|RISE|RMS|SDeViation|TYCInteg|TYINteG|V1|V2}

**Example** :ANALYSIS:WPARAMETER1:TRACE1:AREA1:TYPE BURST  
:ANALYSIS:WPARAMETER1:TRACE1:AREA1:TYPE? -> :ANALYSIS:WPARAMETER1:TRACE1:AREA1:TYPE BURST

**:ANALYSIS:WPARAMETER<x>:TRENd?**

**Function** Queries all settings related to the trend display of the waveform parameter measurement.

**Syntax** :ANALYSIS:WPARAMETER<x>:TRENd?  
<x> = 1 or 2

**Example** :ANALYSIS:WPARAMETER1:TRENd?  
-> :ANALYSIS:WPARAMETER1:TRENd:HRANGE MAIN;HSPAN 1;MEASURE:CURSOR:C1:POSITION 1.000E+00;STATE 1;:ANALYSIS:WPARAMETER1:TRENd:MEASURE:CURSOR:C2:POSITION 2.000E+00;STATE 1;:ANALYSIS:WPARAMETER1:TRENd:MEASURE:CURSOR:DC:STATE 1;:ANALYSIS:WPARAMETER1:TRENd:MEASURE:MODE CURSOR;:ANALYSIS:WPARAMETER1:TRENd:VERTICAL 2.000E+00,1.000E+00

### 5.3 ANALYSIS Group

#### **:ANALYSIS:WPARAMETER<x>:TRENd:**

##### **ASCAle [:EXECute]**

**Function** Executes the auto scaling of the trend display of the waveform parameter measurement.

**Syntax** :ANALYSIS:WPARAMETER<x>:TRENd:  
ASCAle [:EXECute]  
<x> = 1 or 2

**Example** :ANALYSIS:WPARAMETER1:TRENd:ASCALE:  
EXECUTE

#### **:ANALYSIS:WPARAMETER<x>:TRENd:HRANge**

**Function** Sets the target window for trend display of measured waveform parameters or queries the current setting.

**Syntax** :ANALYSIS:WPARAMETER<x>:TRENd:  
HRANge {MAIN|Z1|Z2}  
:ANALYSIS:WPARAMETER<x>:TRENd:HRANge?  
<x> = 1 or 2

**Example** :ANALYSIS:WPARAMETER1:TRENd:HRANGE MAIN  
:ANALYSIS:WPARAMETER1:TRENd:HRANGE?  
-> :ANALYSIS:WPARAMETER1:TRENd:  
HRANGE MAIN

#### **:ANALYSIS:WPARAMETER<x>:TRENd:HSPAN**

**Function** Sets the horizontal span of the trend display of the waveform parameter measurement or queries the current setting.

**Syntax** :ANALYSIS:WPARAMETER<x>:TRENd:HSPAN  
{<NRF>}  
:ANALYSIS:WPARAMETER<x>:TRENd:HSPAN?  
<x> = 1 or 2  
<NRF> = 1 to 100000

**Example** :ANALYSIS:WPARAMETER1:TRENd:HSPAN 1  
:ANALYSIS:WPARAMETER1:TRENd:HSPAN?  
-> :ANALYSIS:WPARAMETER1:TRENd:HSPAN 1

#### **:ANALYSIS:WPARAMETER<x>:TRENd:**

##### **MEASURE?**

**Function** Queries all settings related to the automated measurement of the trend display of the waveform parameter measurement.

**Syntax** :ANALYSIS:WPARAMETER<x>:TRENd:MEASURE?  
<x> = 1 or 2

**Example** :ANALYSIS:WPARAMETER1:TRENd:MEASURE?  
-> :ANALYSIS:WPARAMETER1:TRENd:MEASURE:  
CURSOR:C1:POSITION 1.000E+00;STATE 1;  
:ANALYSIS:WPARAMETER1:TRENd:MEASURE:  
CURSOR:C2:POSITION 2.000E+00;STATE 1;  
:ANALYSIS:WPARAMETER1:TRENd:MEASURE:  
CURSOR:DC:STATE 1;  
:ANALYSIS:WPARAMETER1:TRENd:  
MEASURE:MODE CURSOR

#### **:ANALYSIS:WPARAMETER<x>:TRENd:**

##### **MEASURE:CURSOR?**

**Function** Queries all settings related to the cursor measurement of the trend of the waveform parameter measurement.

**Syntax** :ANALYSIS:WPARAMETER<x>:TRENd:MEASURE:  
CURSOR?  
<x> = 1 or 2

**Example** :ANALYSIS:WPARAMETER1:TRENd:MEASURE:  
CURSOR? -> :ANALYSIS:WPARAMETER1:TRENd:  
MEASURE:CURSOR:C1:POSITION 1.000E+00;  
STATE 1;:ANALYSIS:WPARAMETER1:TRENd:  
MEASURE:CURSOR:C2:POSITION 2.000E+00;  
STATE 1;:ANALYSIS:WPARAMETER1:TRENd:  
MEASURE:CURSOR:DC:STATE 1

#### **:ANALYSIS:WPARAMETER<x>:TRENd:**

##### **MEASURE:CURSOR:ALL**

**Function** Turns ON/OFF all cursors of the trend of the waveform parameter measurement.

**Syntax** :ANALYSIS:WPARAMETER<x>:TRENd:MEASURE:  
CURSOR:ALL {<Boolean>}  
<x> = 1 or 2

**Example** :ANALYSIS:WPARAMETER1:TRENd:MEASURE:  
CURSOR:ALL ON

#### **:ANALYSIS:WPARAMETER<x>:TRENd:**

##### **MEASURE:CURSOR:C<x>?**

**Function** Queries all settings related to the measured value of the cursor of the trend of the waveform parameter measurement.

**Syntax** :ANALYSIS:WPARAMETER<x>:TRENd:MEASURE:  
CURSOR:C<x>?  
<x> of WPARAMETER<x> = 1 or 2  
<x> of C<x> = 1 or 2

**Example** :ANALYSIS:WPARAMETER1:TRENd:MEASURE:  
CURSOR:C1? -> :ANALYSIS:WPARAMETER1:  
TRENd:MEASURE:CURSOR:C1:  
POSITION 1.000E+00;STATE 1

**:ANALysis:WPARAmeter<x>:TRENd:MEASure:CURSor:C<x>:POSition**

**Function** Sets the cursor position of the trend of the waveform parameter measurement or queries the current setting.

**Syntax** :ANALysis:WPARAmeter<x>:TRENd:MEASure:CURSor:C<x>:POSition {<Nrf>}  
:ANALysis:WPARAmeter<x>:TRENd:MEASure:CURSor:C<x>:POSition?  
<x> of WPARAmeter<x> = 1 or 2  
<x> of C<x> = 1 or 2  
<Nrf> = -5 to 5 div

**Example** :ANALYSIS:WPARAMETER1:TREND:MEASURE:  
CURSOR:C1:POSITION 1  
:ANALYSIS:WPARAMETER1:TREND:MEASURE:  
CURSOR:C1:POSITION? -> :ANALYSIS:  
WPARAMETER1:TREND:MEASURE:  
CURSOR:C1:POSITION 1.000E+00  
CURSOR:C1:STATE? -> :ANALYSIS:  
WPARAMETER1:TREND:MEASURE:  
CURSOR:C1:STATE 1

**:ANALysis:WPARAmeter<x>:TRENd:MEASure:CURSor:C<x>:STATe**

**Function** Turns ON/OFF the cursor of the trend of the waveform parameter measurement or queries the current setting.

**Syntax** :ANALysis:WPARAmeter<x>:TRENd:MEASure:CURSor:C<x>:STATe {<Boolean>}  
:ANALysis:WPARAmeter<x>:TRENd:MEASure:CURSor:C<x>:STATe?  
<x> of WPARAmeter<x> = 1 or 2  
<x> of C<x> = 1 or 2

**Example** :ANALYSIS:WPARAMETER1:TREND:MEASURE:  
CURSOR:C1:STATE ON  
:ANALYSIS:WPARAMETER1:TREND:MEASURE:  
CURSOR:C1:STATE? -> :ANALYSIS:  
WPARAMETER1:TREND:MEASURE:  
CURSOR:C1:STATE 1

**:ANALysis:WPARAmeter<x>:TRENd:MEASure:CURSor:C<x>:VALue?**

**Function** Queries the measured value of the cursor of the trend of the waveform parameter measurement.

**Syntax** :ANALysis:WPARAmeter<x>:TRENd:MEASure:CURSor:C<x>:VALue?  
<x> of WPARAmeter<x> = 1 or 2  
<x> of C<x> = 1 or 2

**Example** :ANALYSIS:WPARAMETER1:TREND:MEASURE:  
CURSOR:C1:VALUE? -> :ANALYSIS:  
WPARAMETER1:TREND:MEASURE:  
CURSOR:C1:VALUE 1.000E+00

**:ANALysis:WPARAmeter<x>:TRENd:MEASure:CURSor:DC?**

**Function** Queries all settings related to the measurement between cursors on the trend of the waveform parameter measurement.

**Syntax** :ANALysis:WPARAmeter<x>:TRENd:MEASure:CURSor:DC?  
<x> of WPARAmeter<x> = 1 or 2

**Example** :ANALYSIS:WPARAMETER1:TREND:MEASURE:  
CURSOR:DC? -> :ANALYSIS:WPARAMETER1:  
TREND:MEASURE:CURSOR:DC:STATE 1

**:ANALysis:WPARAmeter<x>:TRENd:MEASure:CURSor:DC:STATe**

**Function** Turns ON/OFF the measurement between cursors on the trend of the waveform parameter measurement or queries the current setting.

**Syntax** :ANALysis:WPARAmeter<x>:TRENd:MEASure:CURSor:DC:STATe {<Boolean>}  
:ANALysis:WPARAmeter<x>:TRENd:MEASure:CURSor:DC:STATe?  
<x> of WPARAmeter<x> = 1 or 2

**Example** :ANALYSIS:WPARAMETER1:TREND:MEASURE:  
CURSOR:DC:STATE ON  
:ANALYSIS:WPARAMETER1:TREND:MEASURE:  
CURSOR:DC:STATE? -> :ANALYSIS:  
WPARAMETER1:TREND:MEASURE:  
CURSOR:DC:STATE 1

**:ANALysis:WPARAmeter<x>:TRENd:MEASure:CURSor:DC:VALue?**

**Function** Sets the measured value between cursors on the trend of the waveform parameter measurement or queries the current setting.

**Syntax** :ANALysis:WPARAmeter<x>:TRENd:MEASure:CURSor:DC:VALue?  
<x> of WPARAmeter<x> = 1 or 2

**Example** :ANALYSIS:WPARAMETER1:TREND:MEASURE:  
CURSOR:DC:VALUE? -> :ANALYSIS:  
WPARAMETER1:TREND:MEASURE:  
CURSOR:DC:VALUE 1.000E+00

### 5.3 ANALYSIS Group

#### **:ANALysis:WPARAMeter<x>:TREND:MEASure:CURSor:LINKage**

**Function** Turns ON/OFF the trend cursor link of the waveform parameter measurement or queries the current setting.

**Syntax** :ANALysis:WPARAMeter<x>:TREND:MEASure:CURSor:LINKage {<Boolean>}  
:ANALysis:WPARAMeter<x>:TREND:MEASure:CURSor:LINKage?  
<x> = 1 or 2

**Example** :ANALYSIS:WPARAMETER1:TREND:MEASURE:CURSOR:LINKAGE ON  
:ANALYSIS:WPARAMETER1:TREND:MEASURE:CURSOR:LINKAGE? -> :ANALYSIS:WPARAMETER1:TREND:MEASURE:CURSOR:LINKAGE 1

#### **:ANALysis:WPARAMeter<x>:TREND:MEASure:MODE**

**Function** Sets the automated measurement mode of the trend of the waveform parameter measurement or queries the current setting.

**Syntax** :ANALysis:WPARAMeter<x>:TREND:MEASure:MODE {CURSor|OFF}  
:ANALysis:WPARAMeter<x>:TREND:MEASure:MODE?  
<x> = 1 or 2

**Example** :ANALYSIS:WPARAMETER1:TREND:MEASURE:MODE CURSOR  
:ANALYSIS:WPARAMETER1:TREND:MEASURE:MODE? -> :ANALYSIS:WPARAMETER1:TREND:MEASURE:MODE CURSOR

#### **:ANALysis:WPARAMeter<x>:TREND:VERTical**

**Function** Sets the vertical range of the trend of the waveform parameter measurement or queries the current setting.

**Syntax** :ANALysis:WPARAMeter<x>:TREND:VERTical {<NRf>,<NRf>}  
:ANALysis:WPARAMeter<x>:TREND:VERTical?  
<x> = 1 or 2  
<NRf> = -1.000E+31 to 1.000E+31 (div)

**Example** :ANALYSIS:WPARAMETER1:TREND:VERTICAL 1,2  
:ANALYSIS:WPARAMETER1:TREND:VERTICAL? -> :ANALYSIS:WPARAMETER1:TREND:VERTICAL 2.000E+00,1.000E+00

#### **:ANALysis:XY<x>?**

**Function** Queries all settings related to the XY display function.

**Syntax** :ANALysis:XY<x>?  
<x> = 1 or 2

**Example** :ANALYSIS:XY1? -> :ANALYSIS:XY1:  
GATE:ALEVEL HIGH;HYSTERESIS1 1.000E+00;  
HYSTERESIS2 1.000E+00;  
HYSTERESIS3 1.000E+00;  
HYSTERESIS4 1.000E+00;  
HYSTERESIS5 1.000E+00;  
HYSTERESIS6 1.000E+00;  
HYSTERESIS7 1.000E+00;  
HYSTERESIS8 1.000E+00;LEVEL1 1.000E+00;  
LEVEL2 1.000E+00;LEVEL3 1.000E+00;  
LEVEL4 1.000E+00;LEVEL5 1.000E+00;  
LEVEL6 1.000E+00;LEVEL7 1.000E+00;  
LEVEL8 1.000E+00;TRACE 1;:ANALYSIS:XY1:  
MEASURE:CURSOR:X1:POSITION 1.000E+00;;  
ANALYSIS:XY1:MEASURE:CURSOR:X2:  
POSITION 2.000E+00;;:ANALYSIS:XY1:  
MEASURE:CURSOR:XLINKAGE 1;Y1:  
POSITION 1.000E+00;;:ANALYSIS:XY1:  
MEASURE:CURSOR:Y2:POSITION 2.000E+00;;  
ANALYSIS:XY1:MEASURE:CURSOR:  
YLINKAGE 1;;:ANALYSIS:XY1:MEASURE:  
MODE CURSOR;XYINTEG:LOOP CLOSE;  
POLARITY CCW;;:ANALYSIS:XY1:  
TRANGE 1.000E+00,0.000E+00;WINDOW MAIN;  
XTRACE 1;YTRACE 1

#### **:ANALysis:XY<x>:GATE?**

**Function** Queries all settings related to the gate function of the XY display.

**Syntax** :ANALysis:XY<x>:GATE?  
<x> = 1 or 2

**Example** :ANALYSIS:XY1:GATE? -> :ANALYSIS:XY1:  
GATE:ALEVEL HIGH;  
HYSTERESIS1 1.000E+00;  
HYSTERESIS2 1.000E+00;  
HYSTERESIS3 1.000E+00;  
HYSTERESIS4 1.000E+00;  
HYSTERESIS5 1.000E+00;  
HYSTERESIS6 1.000E+00;  
HYSTERESIS7 1.000E+00;  
HYSTERESIS8 1.000E+00;LEVEL1 1.000E+00;  
LEVEL2 1.000E+00;LEVEL3 1.000E+00;  
LEVEL4 1.000E+00;LEVEL5 1.000E+00;  
LEVEL6 1.000E+00;LEVEL7 1.000E+00;  
LEVEL8 1.000E+00;TRACE 1

**:ANALYSIS:XY<x>:GATE:ALEVEL**

**Function** Sets the active level of the gate of the XY display or queries the current setting.

**Syntax** :ANALYSIS:XY<x>:GATE:ALEVEL {HIGH|LOW}  
:ANALYSIS:XY<x>:GATE:ALEVEL?  
<x> = 1 or 2

**Example** :ANALYSIS:XY1:GATE:ALEVEL HIGH  
:ANALYSIS:XY1:GATE:ALEVEL?  
-> :ANALYSIS:XY1:GATE:ALEVEL HIGH

**:ANALYSIS:XY<x>:GATE:HYSTERESIS<x>**

**Function** Sets the hysteresis of the gate of the XY display or queries the current setting.

**Syntax** :ANALYSIS:XY<x>:GATE:HYSTERESIS<x>  
{<NRF>}  
:ANALYSIS:XY<x>:GATE:HYSTERESIS<x>?  
<x> of XY<x> = 1 or 2  
<x> of HYSTERESIS<x> = 1 to 8  
<NRF> = 0 to 4 (div)

**Example** :ANALYSIS:XY1:GATE:HYSTERESIS1 1  
:ANALYSIS:XY1:GATE:HYSTERESIS1?  
-> :ANALYSIS:XY1:GATE:  
HYSTERESIS1 1.000E+00

**:ANALYSIS:XY<x>:GATE:LEVEL<x>**

**Function** Sets the level of the gate of the XY display or queries the current setting.

**Syntax** :ANALYSIS:XY<x>:GATE:LEVEL<x> {<NRF>,<V  
oltage>,<Current>}  
:ANALYSIS:XY<x>:GATE:LEVEL<x>?  
<x> of XY<x> = 1 or 2  
<x> of LEVEL<x> = 1 to 8  
<NRF>, <Voltage>, and <Current> = See the DL9000  
User's Manual.

**Example** :ANALYSIS:XY1:GATE:LEVEL1 1  
:ANALYSIS:XY1:GATE:LEVEL1?  
-> :ANALYSIS:XY1:GATE:LEVEL1 1.000E+00

**:ANALYSIS:XY<x>:GATE:TRACE**

**Function** Sets the gate trace of the XY display or queries the current setting.

**Syntax** :ANALYSIS:XY<x>:GATE:TRACE {<NRF>|OFF}  
:ANALYSIS:XY<x>:GATE:TRACE?  
<x> = 1 or 2  
<NRF> = 1 to 8

**Example** :ANALYSIS:XY1:GATE:TRACE 1  
:ANALYSIS:XY1:GATE:TRACE? -> :ANALYSIS:  
XY1:GATE:TRACE 1

**:ANALYSIS:XY<x>:MEASURE?**

**Function** Queries all settings related to the automated measurement of the XY display.

**Syntax** :ANALYSIS:XY<x>:MEASURE?  
<x> = 1 or 2

**Example** :ANALYSIS:XY1:MEASURE? -> :ANALYSIS:  
XY1:MEASURE:CURSOR:X1:  
POSITION 1.000E+00;:ANALYSIS:XY1:  
MEASURE:CURSOR:X2:POSITION 2.000E+00;:  
ANALYSIS:XY1:MEASURE:CURSOR:XLINKAGE 1;  
Y1:POSITION 1.000E+00;:ANALYSIS:XY1:  
MEASURE:CURSOR:Y2:POSITION 2.000E+00;:  
ANALYSIS:XY1:MEASURE:CURSOR:  
YLINKAGE 1;:ANALYSIS:XY1:MEASURE:  
MODE CURSOR;XYINTEG:LOOP CLOSE;  
POLARITY CCW

**:ANALYSIS:XY<x>:MEASURE:CURSOR?**

**Function** Queries all settings related to the cursor measurement of the XY display.

**Syntax** :ANALYSIS:XY<x>:MEASURE:CURSOR?  
<x> = 1 or 2

**Example** :ANALYSIS:XY1:MEASURE:CURSOR?  
-> :ANALYSIS:XY1:MEASURE:CURSOR:X1:  
POSITION 1.000E+00;:ANALYSIS:XY1:  
MEASURE:CURSOR:X2:POSITION 2.000E+00;:  
ANALYSIS:XY1:MEASURE:CURSOR:XLINKAGE 1;  
Y1:POSITION 1.000E+00;:ANALYSIS:XY1:  
MEASURE:CURSOR:Y2:POSITION 2.000E+00;:  
ANALYSIS:XY1:MEASURE:CURSOR:YLINKAGE 1

### 5.3 ANALYSIS Group

#### **:ANALYSIS:XY<x>:MEASURE:Cursors:XLINKage**

**Function** Turns ON/OFF the horizontal cursor link on the XY display or queries the current setting.

**Syntax** :ANALYSIS:XY<x>:MEASURE:Cursors:  
XLINKage {<Boolean>}  
:ANALYSIS:XY<x>:MEASURE:Cursors:  
XLINKage?  
<x> = 1 or 2

**Example** :ANALYSIS:XY1:MEASURE:Cursors:  
XLINKAGE ON  
:ANALYSIS:XY1:MEASURE:Cursors:XLINKAGE?  
-> :ANALYSIS:XY1:MEASURE:Cursors:  
XLINKAGE 1

#### **:ANALYSIS:XY<x>:MEASURE:Cursors:X<x>?**

**Function** Queries all settings related to the horizontal cursor of the XY display.

**Syntax** :ANALYSIS:XY<x>:MEASURE:Cursors:X<x>?  
<x> of XY<x> = 1 or 2  
<x> of X<x> = 1 or 2

**Example** :ANALYSIS:XY1:MEASURE:Cursors:X1?  
-> :ANALYSIS:XY1:MEASURE:Cursors:X1:  
POSITION 1.000E+00

#### **:ANALYSIS:XY<x>:MEASURE:Cursors:X<x>:POSITION**

**Function** Sets the horizontal cursor position of the XY display or queries the current setting.

**Syntax** :ANALYSIS:XY<x>:MEASURE:Cursors:X<x>:  
POSITION {<NRf>}  
:ANALYSIS:XY<x>:MEASURE:Cursors:X<x>:  
POSITION?  
<x> of XY<x> = 1 or 2  
<x> of X<x> = 1 or 2  
<NRf> = -4 to 4 (div)

**Example** :ANALYSIS:XY1:MEASURE:Cursors:X1:  
POSITION 1  
:ANALYSIS:XY1:MEASURE:Cursors:X1:  
POSITION? -> :ANALYSIS:XY1:MEASURE:  
Cursors:X1:POSITION 1.000E+00

#### **:ANALYSIS:XY<x>:MEASURE:Cursors:X<x>:VALUE?**

**Function** Queries the voltage value at the horizontal cursor of the XY display.

**Syntax** :ANALYSIS:XY<x>:MEASURE:Cursors:X<x>:  
VALUE?  
<x> of XY<x> = 1 or 2  
<x> of X<x> = 1 or 2

**Example** :ANALYSIS:XY1:MEASURE:Cursors:X1:VALUE?  
-> :ANALYSIS:XY1:MEASURE:Cursors:X1:  
VALUE 1.000E+00

#### **:ANALYSIS:XY<x>:MEASURE:Cursors:YLINKage**

**Function** Turns ON/OFF the vertical cursor link on the XY display or queries the current setting.

**Syntax** :ANALYSIS:XY<x>:MEASURE:Cursors:  
YLINKage {<Boolean>}  
:ANALYSIS:XY<x>:MEASURE:Cursors:  
YLINKage?  
<x> = 1 or 2

**Example** :ANALYSIS:XY1:MEASURE:Cursors:  
YLINKAGE ON  
:ANALYSIS:XY1:MEASURE:Cursors:YLINKAGE?  
-> :ANALYSIS:XY1:MEASURE:Cursors:  
YLINKAGE 1

#### **:ANALYSIS:XY<x>:MEASURE:Cursors:Y<x>?**

**Function** Queries all settings related to the vertical cursor of the XY display.

**Syntax** :ANALYSIS:XY<x>:MEASURE:Cursors:Y<x>?  
<x> of XY<x> = 1 or 2  
<x> of Y<x> = 1 or 2

**Example** :ANALYSIS:XY1:MEASURE:Cursors:Y1?  
-> :ANALYSIS:XY1:MEASURE:Cursors:Y1:  
POSITION 1.000E+00

#### **:ANALYSIS:XY<x>:MEASURE:Cursors:Y<x>:POSITION**

**Function** Sets the vertical cursor position of the XY display or queries the current setting.

**Syntax** :ANALYSIS:XY<x>:MEASURE:Cursors:Y<x>:  
POSITION {<NRf>}  
:ANALYSIS:XY<x>:MEASURE:Cursors:Y<x>:  
POSITION?  
<x> of XY<x> = 1 or 2  
<x> of Y<x> = 1 or 2  
<NRf> = -4 to 4 (div)

**Example** :ANALYSIS:XY1:MEASURE:Cursors:Y1:  
POSITION 1  
:ANALYSIS:XY1:MEASURE:Cursors:Y1:  
POSITION? -> :ANALYSIS:XY1:MEASURE:  
Cursors:Y1:POSITION 1.000E+00



**:ANALysis:XY<x>:MEASure:CURSor:Y<x>:VALue?**

Function Queries the voltage value at the vertical cursor of the XY display.

Syntax :ANALysis:XY<x>:MEASure:CURSor:Y<x>:VALue?

<x> of XY<x> = 1 or 2

<x> of Y<x> = 1 or 2

Example :ANALYSIS:XY1:MEASURE:CURSOR:Y1:VALUE?  
-> :ANALYSIS:XY1:MEASURE:CURSOR:Y1:VALUE 1.000E+00

**:ANALysis:XY<x>:MEASure:MODE**

Function Sets the automated measurement mode of the XY display or queries the current setting.

Syntax :ANALysis:XY<x>:MEASure:MODE {CURSor|OFF|XYINteG}

:ANALysis:XY<x>:MEASure:MODE?

<x> = 1 or 2

Example :ANALYSIS:XY1:MEASURE:MODE CURSOR  
:ANALYSIS:XY1:MEASURE:MODE?  
-> :ANALYSIS:XY1:MEASURE:MODE CURSOR

**:ANALysis:XY<x>:MEASure:XYINteG?**

Function Queries all settings related to the integration of the XY display.

Syntax :ANALysis:XY<x>:MEASure:XYINteG?

<x> = 1 or 2

Example :ANALYSIS:XY1:MEASURE:XYINTEG?  
-> :ANALYSIS:XY1:MEASURE:XYINTEG:LOOP CLOSE;POLARITY CCW

**:ANALysis:XY<x>:MEASure:XYINteG:LOOP**

Function Sets the integration mode of the XY display or queries the current setting.

Syntax :ANALysis:XY<x>:MEASure:XYINteG:LOOP {CLoSe|OPeN}

:ANALysis:XY<x>:MEASure:XYINteG:LOOP?

<x> = 1 or 2

Example :ANALYSIS:XY1:MEASURE:XYINTEG:LOOP CLOSE  
:ANALYSIS:XY1:MEASURE:XYINTEG:LOOP?  
-> :ANALYSIS:XY1:MEASURE:XYINTEG:LOOP CLOSE

**:ANALysis:XY<x>:MEASure:XYINteG:****POLarity**

Function Sets the integration polarity of the XY display or queries the current setting.

Syntax :ANALysis:XY<x>:MEASure:XYINteG:

POLarity {CCW|CW}

:ANALysis:XY<x>:MEASure:XYINteG:

POLarity?

<x> = 1 or 2

Example :ANALYSIS:XY1:MEASURE:XYINTEG:POLARITY CCW  
:ANALYSIS:XY1:MEASURE:XYINTEG:POLARITY?  
-> :ANALYSIS:XY1:MEASURE:XYINTEG:POLARITY CCW

**:ANALysis:XY<x>:MEASure:XYINteG:****VALue?**

Function Queries the integral value of the XY display.

Syntax :ANALysis:XY<x>:MEASure:XYINteG:VALue?

<x> = 1 or 2

Example :ANALYSIS:XY1:MEASURE:XYINTEG:VALUE?  
-> :ANALYSIS:XY1:MEASURE:XYINTEG:VALUE 1.000E+00

**:ANALysis:XY<x>:TRANge (Time Range)**

Function Sets the measurement range of the XY display or queries the current setting.

Syntax :ANALysis:XY<x>:TRANge {<NRf>,<NRf>}

:ANALysis:XY<x>:TRANge?

<x> = 1 or 2

<NRf> = -5 to 5 div

**:ANALysis:XY<x>:WINDow**

Function Sets the measurement source window of the XY display or queries the current setting.

Syntax :ANALysis:XY<x>:WINDow {MAIN|Z1|Z2}

:ANALysis:XY<x>:WINDow?

<x> = 1 or 2

Example :ANALYSIS:XY1:WINDOW MAIN  
:ANALYSIS:XY1:WINDOW? -> :ANALYSIS:XY1:WINDOW MAIN

### 5.3 ANALysis Group

---

#### **:ANALysis:XY<x>:XTRace**

**Function** Sets the X-axis trace of the XY display or queries the current setting.

**Syntax** :ANALysis:XY<x>:XTRace {<NRf>}  
:ANALysis:XY<x>:XTRace?  
<x> = 1 or 2  
<NRf> = 1 to 8

**Example** :ANALYSIS:XY1:XTRACE 1  
:ANALYSIS:XY1:XTRACE? -> :ANALYSIS:XY1:  
XTRACE 1

#### **:ANALysis:XY<x>:YTRace**

**Function** Sets the Y-axis trace of the XY display or queries the current setting.

**Syntax** :ANALysis:XY<x>:YTRace {<NRf>}  
:ANALysis:XY<x>:YTRace?  
<x> = 1 or 2  
<NRf> = 1 to 8

**Example** :ANALYSIS:XY1:YTRACE 1  
:ANALYSIS:XY1:YTRACE? -> :ANALYSIS:XY1:  
YTRACE 1

---

## 5.4 ASETup Group

### **:ASETup:EXECute**

Function Execute auto setup.

Syntax :ASETup:EXECute

Example :ASETUP:EXECUTE

### **:ASETup:UNDO**

Function Cancels auto setup that has been executed.

Syntax :ASETup:UNDO

Example :ASETUP:UNDO

---

## 5.5 CALibrate Group

### **:CALibrate?**

Function Queries all settings related to the calibration.

Syntax :CALibrate?

Example :CALIBRATE? -> :CALIBRATE:MODE AUTO

### **:CALibrate:EXECute**

Function Executes calibration.

Syntax :CALibrate:EXECute

Example :CALIBRATE:EXECUTE

### **:CALibrate:MODE**

Function Turns ON/OFF the auto calibration or queries the current setting.

Syntax :CALibrate:MODE {AUTO|OFF}

:CALibrate:MODE?

Example :CALIBRATE:MODE AUTO

:CALIBRATE:MODE?

-> :CALIBRATE:MODE AUTO

## 5.6 CHANnel Group

### **:CHANnel<x>?**

**Function** Queries all settings related to the channel.

**Syntax** :CHANnel<x>?  
<x> = 1 to 4

**Example** :CHANNEL1? -> :CHANNEL1:SELECT INPUT;  
DISPLAY 1;BWIDTh FULL;COUPLING DC;  
DESKEW 0.000E+00;INVERT 0;LABEL:  
DEFINE "CH1";MODE 1;:CHANNEL1:  
OCANCEL 0;OFFSEt 0.000E+00;  
POSITION 0.000E+00;PROBE:MODE 1;:  
CHANNEL1:SVALUE 0;VDIV 1.000E+00

### **:CHANnel<x>:ASCaLe[:EXECute]**

**Function** Executes the auto scaling of the channel.

**Syntax** CHANnel<x>:ASCaLe[:EXECute]  
<x> = 1 to 4

**Example** CHANNEL1:ASCALE:EXECUTE

### **:CHANnel<x>:BWIDTh**

**Function** Sets the input filter of the channel or queries the current setting.

**Syntax** :CHANnel<x>:BWIDTh {<Frequency>|FULL}  
:CHANnel<x>:BWIDTh?  
<x> = 1 to 4  
<Frequency> = See the DL9000 User's Manual

**Example** :CHANNEL1:BWIDTh FULL  
:CHANNEL1:BWIDTh? -> :CHANNEL1:  
BWIDTh FULL

### **:CHANnel<x>:COUPLing**

**Function** Sets the input coupling of the channel or queries the current setting.

**Syntax** :CHANnel<x>:COUPLing {AC|DC|DC50|GND}  
:CHANnel<x>:COUPLing?  
<x> = 1 to 4

**Example** :CHANNEL1:COUPLING GND  
:CHANNEL1:COUPLING? -> :CHANNEL1:  
COUPLING GND

### **:CHANnel<x>:DESKew**

**Function** Sets the skew correction of the channel or queries the current setting.

**Syntax** :CHANnel<x>:DESKew {<Time>}  
:CHANnel<x>:DESKew?  
<x> = 1 to 4  
<Time> = -80 ns to 80 ns in 10-ps steps

**Example** :CHANNEL1:DESKEW 1NS  
:CHANNEL1:DESKEW?  
-> :CHANNEL1:DESKEW 1.000E-09

### **:CHANnel<x>:DISPlay**

**Function** Turns ON/OFF the display of the channel or queries the current setting.

**Syntax** :CHANnel<x>:DISPlay {<Boolean>}  
:CHANnel<x>:DISPlay?  
<x> = 1 to 4

**Example** :CHANNEL1:DISPlay ON  
:CHANNEL1:DISPlay?  
-> :CHANNEL1:DISPlay 1

### **:CHANnel<x>:INVert**

**Function** Turns ON/OFF the inverted display of the channel or queries the current setting.

**Syntax** :CHANnel<x>:INVert {<Boolean>}  
:CHANnel<x>:INVert?  
<x> = 1 to 4

**Example** :CHANNEL1:INVERT ON  
:CHANNEL1:INVERT -> :CHANNEL1:INVERT 1

### **:CHANnel<x>:LABel?**

**Function** Queries all settings related to the waveform label of the channel.

**Syntax** :CHANnel<x>:LABel?  
<x> = 1 to 4

**Example** :CHANNEL1:LABel? -> :CHANNEL1:LABel:  
DEFINE "CH1";MODE 0

### **:CHANnel<x>:LABel[:DEFine]**

**Function** Sets the waveform label of channel or queries the current setting.

**Syntax** :CHANnel<x>:LABel[:DEFine] {<String>}  
:CHANnel<x>:LABel[:DEFine]?  
<x> = 1 to 4  
<String> = Up to 8 characters

**Example** :CHANNEL1:LABel:DEFINE "CH1"  
:CHANNEL1:LABel:DEFINE? -> :CHANNEL1:  
LABel:DEFINE "CH1"

### **:CHANnel<x>:LABel:MODE**

**Function** Turns ON/OFF the waveform label display of the channel or queries the current setting.

**Syntax** :CHANnel<x>:LABel:MODE {<Boolean>}  
:CHANnel<x>:LABel:MODE?  
<x> = 1 to 4

**Example** :CHANNEL1:LABel:MODE ON  
:CHANNEL1:LABel:MODE? -> :CHANNEL1:  
LABel:MODE 1

## 5.6 CHANnel Group

### **:CHANnel<x>:OCANcel**

**Function** Turns ON/OFF the offset cancel of the channel or queries the current setting.

**Syntax** :CHANnel<x>:OCANcel {<Boolean>}  
:CHANnel<x>:OCANcel?  
<x> = 1 to 4

**Example** :CHANNEL1:OCANCEL ON  
:CHANNEL1:OCANCEL? -> :CHANNEL1:  
OCANCEL 1

### **:CHANnel<x>:OFFSet**

**Function** Sets the offset voltage of the channel or queries the current setting.

**Syntax** :CHANnel<x>:OFFSet  
{<Voltage>|<Current>}  
:CHANnel<x>:OFFSet?  
<x> = 1 to 4  
<Voltage> and <Current> = See the DL9000 User's Manual.

**Example** :CHANNEL1:OFFSET 1V  
:CHANNEL1:OFFSET? -> :CHANNEL1:  
OFFSET 1.000E+00

### **:CHANnel<x>:POSition**

**Function** Sets the vertical position of the channel or queries the current setting.

**Syntax** :CHANnel<x>:POSition {<NRf>}  
:CHANnel<x>:POSition?  
<x> = 1 to 4  
<NRf> = -4 to 4 (div)

**Example** :CHANNEL1:POSITION 1  
:CHANNEL1:POSITION? -> :CHANNEL1:  
POSITION 1.000E+00

### **:CHANnel<x>:PROBe?**

**Function** Queries all settings related to the probe attenuation of the channel.

**Syntax** :CHANnel<x>:PROBe?  
<x> = 1 to 4

**Example** :CHANNEL1:PROBE? -> :CHANNEL1:PROBE:  
MODE 1

### **:CHANnel<x>:PROBe[:MODE]**

**Function** Sets the probe attenuation of the channel or queries the current setting.

**Syntax** :CHANnel<x>:PROBe[:MODE] {<NRf>|AUTO|  
C1|C10|C100}  
:CHANnel<x>:PROBe[:MODE]?  
<x> = 1 to 4  
<NRf> = 1, 2, 5, 10, 20, 50, 100, 200, 500, or 1000

**Example** :CHANNEL1:PROBE:MODE 1  
:CHANNEL1:PROBE:MODE? -> :CHANNEL1:  
PROBE:MODE 1

### **:CHANnel<x>:PROBe:AUTO?**

**Function** Queries the probe attenuation of the channel when set to AUTO.

**Syntax** :CHANnel<x>:PROBe:AUTO?  
<x> = 1 to 4

**Example** :CHANNEL1:PROBE:AUTO? -> :CHANNEL1:  
PROBE:AUTO 1

### **:CHANnel<x>:PZCalibrate[:EXECute]**

**Function** Execute the zero adjustment of the current probe of the channel.

**Syntax** :CHANnel<x>:PZCalibrate[:EXECute]  
<x> = 1 to 4

**Example** :CHANNEL1:PZCALIBRATE:EXECUTE

### **:CHANnel<x>:SElect**

**Function** Sets the waveform (input/computation) to be assigned to the input channel or queries the current setting.

**Syntax** :CHANnel<x>:SElect {INPut|MATH}  
:CHANnel<x>:SElect?  
<x> = 1 to 4

**Example** :CHANNEL1:SELECT INPUT  
:CHANNEL1:SELECT? -> :CHANNEL1:SELECT  
INPUT

### **:CHANnel<x>:SVALue (Scale VALUE)**

**Function** Turns ON/OFF the scale display of the channel or queries the current setting.

**Syntax** :CHANnel<x>:SVALue {<Boolean>}  
:CHANnel<x>:SVALue?  
<x> = 1 to 4

**Example** :CHANNEL1:SVALUE ON  
:CHANNEL1:SVALUE? -> :CHANNEL1:SVALUE 1

### **:CHANnel<x>:VDIV**

**Function** Sets the vertical sensitivity (V/div) of the channel or queries the current setting.

**Syntax** :CHANnel<x>:VDIV {<Voltage>|<Current>}  
:CHANnel<x>:VDIV?  
<x> = 1 to 4  
<Voltage> and <Current> = See the DL9000 User's Manual.

**Example** :CHANNEL1:VDIV 5V  
:CHANNEL1:VDIV? -> :CHANNEL1:  
VDIV 5.000E+00

---

## 5.7 CLEar Group

### **:CLEar:ACCumulate**

Function Clears accumulated waveforms.

Syntax :CLEar:ACCumulate

Example :CLEAR:ACCUMULATE

### **:CLEar[:HISTory]**

Function Clears history waveforms.

Syntax :CLEar[:HISTory]

Example :CLEAR:HISTORY

### **:CLEar:SNAP**

Function Clears snapshot waveforms.

Syntax :CLEar:SNAP

Example :CLEAR:SNAP

## 5.8 COMMunicate Group

The commands in this group deal with communications. There are no front panel keys that correspond to the commands in this group.

### :COMMunicate?

Function Queries all settings related to communications.

Syntax :COMMunicate?

Example :COMMUNICATE? -> :COMMUNICATE:  
HEADER 1;OPSE 352;OVERLAP 352;VERBOSE 1

### :COMMunicate:HEADer

Function Sets whether to add a header to the response to a query (example CHANNEL1:VOLTAGE:PROBE 10) or not add the header (example 10) or queries the current setting.

Syntax :COMMunicate:HEADer {<Boolean>}  
:COMMunicate:HEADer?

Example :COMMUNICATE:HEADER ON  
:COMMUNICATE:HEADER? -> :COMMUNICATE:  
HEADER 1

### :COMMunicate:LOCKout

Function Sets or clears local lockout.

Syntax :COMMunicate:LOCKout {<Boolean>}  
:COMMunicate:LOCKout?

Example :COMMUNICATE:LOCKOUT ON  
:COMMUNICATE:LOCKOUT? -> :COMMUNICATE:  
LOCKOUT 1

Description This command is dedicated to the USB and Ethernet interface (option).

### :COMMunicate:OPSE (Operation Pending Status Enable register)

Function Sets the overlap command that is used by the \*OPC, \*OPC?, and \*WAI commands or queries the current setting.

Syntax :COMMunicate:OPSE <Register>  
:COMMunicate:OPSE?  
<Register> = 0 to 65535, :COMMunicate:OVERlap  
See the command diagram.

Example :COMMUNICATE:OPSE 65535  
:COMMUNICATE:OPSE? -> :COMMUNICATE:  
OPSE 2400

Description In the above example, all bits are set to 1 to make all overlap commands applicable. However, bits fixed to 0 are not set to 1. Thus, the response to the query indicates 1 for bits 5, 6, 8, and 11 only.

### :COMMunicate:OPSR? (Operation Pending Status Register)

Function Queries the value of the operation pending status register.

Syntax :COMMunicate:OPSR?

Example :COMMUNICATE:OPSR? -> 0

Description For details on the operation pending status register, see the figure for the :COMMunicate:OVERlap command.

### :COMMunicate:OVERlap

Function Sets the commands to operate as overlap commands or queries the current setting.

Syntax :COMMunicate:OVERlap <Register>  
:COMMunicate:OVERlap?  
<Register> = 0 to 65535

Example :COMMUNICATE:OVERLAP 65535  
:COMMUNICATE:OVERLAP? -> :COMMUNICATE:  
OVERLAP 2400

Description

- In the above example, all bits are set to 1 to make all overlap commands applicable. However, bits fixed to 0 are not set to 1. Thus, the response to the query indicates 1 for bits 5, 6, 8, and 11 only.
- For the description of how to synchronize the program using :COMMunicate:OVERlap, see page 4-7.
- In the above example, bits 5, 6, 8, and 11 are set to 1 to make all overlap commands applicable.

Operation pending status register/overlap enable register

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	SCH	0	0	HST	0	ACS	PRN	0	0	0	0	0

When bit 5 (PRN) = 1:

Built-in printer operation not complete

When bit 6 (ACS) = 1:

Access to the medium not complete.

When bit 8 (HST) = 1:

History search execution not complete

When bit 11 (SCH) = 1:

Search execution not complete



**:COMMunicate:REMOte**

Function Sets remote or local. ON is remote mode.

Syntax :COMMunicate:REMOte {<Boolean>}  
:COMMunicate:REMOte?

Example :COMMUNICATE:REMOTE ON  
:COMMUNICATE:REMOTE? -> :COMMUNICATE:  
REMOTE 1

Description This command is dedicated to the USB and Ethernet interface (option).

**:COMMunicate:STATus?**

Function Queries line-specific status.

Syntax :COMMunicate:STATus?

Example :COMMUNICATE:STATUS? -> :COMMUNICATE:  
STATUS 0

Description The meaning of each status bit is as follows:

Bit	GP-IB
0	Unrecoverable transmission error
1	Always 0
2	Always 0
3 or greater	Always 0

For USB and Ethernet communications, 0 is always returned.

The status bit is set when the corresponding cause occurs and cleared when it is read.

**:COMMunicate:VERBOse**

Function Sets whether to return the response to a query using full spelling (example CHANNEL1:VOLTAGE:PROBE 10) or using abbreviation (example CHAN:PROB 10) or queries the current setting.

Syntax :COMMunicate:VERBOse {<Boolean>}  
:COMMunicate:VERBOse?

Example :COMMUNICATE:VERBOSE ON  
:COMMUNICATE:VERBOSE? -> :COMMUNICATE:  
VERBOSE 1

**:COMMunicate:WAIT**

Function Waits for one of the specified extended events to occur.

Syntax :COMMunicate:WAIT <Register>  
<Register> = 0 to 65535 (extended event register, see page 6-5.)

Example :COMMUNICATE:WAIT 65535

Description For the description of how to synchronize the program using :COMMunicate:WAIT, see page 4-8.

**:COMMunicate:WAIT?**

Function Creates the response that is returned when the specified event occurs.

Syntax :COMMunicate:WAIT? <Register>  
<Register> = 0 to 65535 (extended event register, see page 6-5.)

Example :COMMUNICATE:WAIT? 65535 -> 1

## 5.9 CURSor Group

### **:CURSor?**

**Function** Queries all settings related to the cursor measurement.

**Syntax** :CURSor?

**Example** :CURSOR? -> :CURSOR:DISPLAY 1;  
HORIZONTAL:BASIC:DV:STATE 1;;CURSOR:  
HORIZONTAL:BASIC:LINKAGE 1;V1:  
POSITION 1.000E+00;STATE 0;;CURSOR:  
HORIZONTAL:BASIC:V2:POSITION 1.000E+00;  
STATE 0;;CURSOR:HORIZONTAL:CALCULATION:  
DEFINE1 "V1";DEFINE2 "V2";DEFINE3 "V1";  
DEFINE4 "V2";STATE1 0;STATE2 0;  
STATE3 0;STATE4 0;;CURSOR:HORIZONTAL:  
TRACE 1;;CURSOR:MARKER:CALCULATION:  
DEFINE1 "T2";DEFINE2 "V2";DEFINE3 "V3";  
DEFINE4 "V4";STATE1 0;STATE2 0;  
STATE3 0;STATE4 0;;CURSOR:MARKER:CM1:  
DT2:STATE 1;;CURSOR:MARKER:CM1:DT3:  
STATE 1;;CURSOR:MARKER:CM1:DT4:  
STATE 1;;CURSOR:MARKER:CM1:DV2:  
STATE 1;;CURSOR:MARKER:CM1:DV3:  
STATE 1;;CURSOR:MARKER:CM1:DV4:  
STATE 1;;CURSOR:MARKER:CM1:  
POSITION 1.000E+00;T:STATE 1;;CURSOR:  
MARKER:CM1:TRACE 1;V:STATE 1;;CURSOR:  
MARKER:CM2:DT1:STATE 1;;CURSOR:MARKER:  
CM2:DT3:STATE 1;;CURSOR:MARKER:CM2:DT4:  
STATE 1;;CURSOR:MARKER:CM2:DV1:  
STATE 1;;CURSOR:MARKER:CM2:DV3:  
STATE .....

### **:CURSor:DISPlay**

**Function** Turns ON/OFF the cursor or queries the current setting.

**Syntax** :CURSor:DISPlay {<Boolean>}  
:CURSor:DISPlay?

**Example** :CURSOR:DISPLAY ON  
:CURSOR:DISPLAY? -> :CURSOR:DISPLAY 1

### **:CURSor:HORizontal?**

**Function** Queries all settings related to the horizontal cursors.

**Syntax** :CURSor:HORizontal?

**Example** :CURSOR:HORIZONTAL? -> :CURSOR:  
HORIZONTAL:BASIC:DV:STATE 1;;  
CURSOR:HORIZONTAL:BASIC:LINKAGE 1;V1:  
POSITION 1.000E+00;STATE 0;;CURSOR:  
HORIZONTAL:BASIC:V2:POSITION 1.000E+00;  
STATE 0;;CURSOR:HORIZONTAL:  
CALCULATION:DEFINE1 "V1";DEFINE2 "V2";  
DEFINE3 "V1";DEFINE4 "V2";STATE1 0;  
STATE2 0;STATE3 0;STATE4 0;;CURSOR:  
HORIZONTAL:TRACE 1

### **:CURSor:HORizontal:BASic?**

**Function** Queries all settings related to basic items of the horizontal cursors.

**Syntax** :CURSor:HORizontal:BASic?

**Example** :CURSOR:HORIZONTAL:BASIC? -> :CURSOR:  
HORIZONTAL:BASIC:DV:STATE 1;;  
CURSOR:HORIZONTAL:BASIC:LINKAGE 1;V1:  
POSITION 1.000E+00;STATE 0;;CURSOR:  
HORIZONTAL:BASIC:V2:POSITION 1.000E+00;  
STATE 0

### **:CURSor:HORizontal[:BASic]:ALL**

**Function** Turns ON/OFF all basic items of the horizontal cursors.

**Syntax** :CURSor:HORizontal[:BASic]:ALL  
{<Boolean>}

**Example** :CURSOR:HORIZONTAL:BASIC:ALL ON

### **:CURSor:HORizontal[:BASic]:DV?**

**Function** Queries all settings related to the  $\Delta V$  measurement of the horizontal cursors.

**Syntax** :CURSor:HORizontal[:BASic]:DV?

**Example** :CURSOR:HORIZONTAL:BASIC:DV?  
-> :CURSOR:HORIZONTAL:BASIC:DV:STATE 1

### **:CURSor:HORizontal[:BASic]:DV:STATE**

**Function** Turns ON/OFF the  $\Delta V$  measurement of the horizontal cursors or queries the current setting.

**Syntax** :CURSor:HORizontal[:BASic]:DV:STATE  
{<Boolean>}  
:CURSor:HORizontal[:BASic]:DV:STATE?

**Example** :CURSOR:HORIZONTAL:BASIC:DV:STATE ON  
:CURSOR:HORIZONTAL:BASIC:DV:STATE?  
-> :CURSOR:HORIZONTAL:BASIC:DV:STATE 1

### **:CURSor:HORizontal[:BASic]:DV:VALue?**

**Function** Queries the voltage value between the horizontal cursors.

**Syntax** :CURSor:HORizontal[:BASic]:DV:VALue?

**Example** :CURSOR:HORIZONTAL:BASIC:DV:VALue?  
-> :CURSOR:HORIZONTAL:BASIC:DV:  
VALUE 1.000E+00

### **:CURSor:HORizontal[:BASic]:LINKage**

**Function** Turns ON/OFF the horizontal cursor link or queries the current setting.

**Syntax** :CURSor:HORizontal[:BASic]:  
LINKage {<Boolean>}  
:CURSor:HORizontal[:BASic]:LINKage?

**Example** :CURSOR:HORIZONTAL:BASIC:LINKAGE ON  
:CURSOR:HORIZONTAL:BASIC:LINKAGE?  
-> :CURSOR:HORIZONTAL:BASIC:LINKAGE 1

**:CURSOR:HORIZONTAL[:BASIC]:V<x>?**

Function Queries all settings related to the horizontal cursor.

Syntax :CURSOR:HORIZONTAL[:BASIC]:V<x>?  
<x> = 1 or 2

Example :CURSOR:HORIZONTAL:BASIC:V1?  
-> :CURSOR:HORIZONTAL:BASIC:V1:POSITION  
1.000E+00;STATE 1

**:CURSOR:HORIZONTAL[:BASIC]:V<x>:JUMP**

Function Jumps the horizontal cursor to the center position of the zoom waveform.

Syntax :CURSOR:HORIZONTAL[:BASIC]:V<x>:  
JUMP {Z1|Z2}  
<x>=1 or 2

Example :CURSOR:HORIZONTAL:BASIC:V1:JUMP Z1

**:CURSOR:HORIZONTAL[:BASIC]:V<x>:POSITION**

Function Sets the horizontal cursor position or queries the current setting.

Syntax :CURSOR:HORIZONTAL[:BASIC]:V<x>:  
POSITION {<NRf>}  
:CURSOR:HORIZONTAL[:BASIC]:V<x>:  
POSITION?  
<x> = 1 or 2  
<NRf> = -4 to 4 (div)

Example :CURSOR:HORIZONTAL:BASIC:V1:POSITION 1  
:CURSOR:HORIZONTAL:BASIC:V1:POSITION?  
-> :CURSOR:HORIZONTAL:BASIC:V1:  
POSITION 1.000E+00

**:CURSOR:HORIZONTAL[:BASIC]:V<x>:STATE**

Function Turns ON/OFF the horizontal cursor or queries the current setting.

Syntax :CURSOR:HORIZONTAL[:BASIC]:V<x>:STATE  
{<Boolean>}  
:CURSOR:HORIZONTAL[:BASIC]:V<x>:STATE?  
<x> = 1 or 2

Example :CURSOR:HORIZONTAL:BASIC:V1:STATE ON  
:CURSOR:HORIZONTAL:BASIC:V1:STATE?  
-> :CURSOR:HORIZONTAL:BASIC:V1:STATE 1

**:CURSOR:HORIZONTAL[:BASIC]:V<x>:VALUE?**

Function Queries the voltage value at the horizontal cursor.

Syntax :CURSOR:HORIZONTAL[:BASIC]:V<x>:VALUE?  
<x> = 1 or 2

Example :CURSOR:HORIZONTAL:BASIC:V1:VALUE?  
-> :CURSOR:HORIZONTAL:BASIC:V1:  
VALUE 1.000E+00

**:CURSOR:HORIZONTAL:CALCULATION?**

Function Queries all settings related to calculation items of the horizontal cursors.

Syntax :CURSOR:HORIZONTAL:CALCULATION?

Example :CURSOR:HORIZONTAL:CALCULATION?  
-> :CURSOR:HORIZONTAL:CALCULATION:  
DEFINE1 "V1";DEFINE2 "V2";DEFINE3 "V1";  
DEFINE4 "V2";STATE1 0;STATE2 0;  
STATE3 0;STATE4 0

**:CURSOR:HORIZONTAL:CALCULATION:ALL**

Function Turns ON/OFF all calculation items of the horizontal cursors.

Syntax :CURSOR:HORIZONTAL:CALCULATION:ALL  
{<Boolean>}

Example :CURSOR:HORIZONTAL:CALCULATION:ALL ON

**:CURSOR:HORIZONTAL:CALCULATION:DEFINE<x>**

Function Sets the equation of the calculation item of the horizontal cursor or queries the current setting.

Syntax :CURSOR:HORIZONTAL:CALCULATION:  
DEFINE<x> {<String>}  
:CURSOR:HORIZONTAL:CALCULATION:  
DEFINE<x>?  
<x> = 1 to 4  
<String> = Up to 128 characters

Example :CURSOR:HORIZONTAL:CALCULATION:  
DEFINE1 "V1"  
:CURSOR:HORIZONTAL:CALCULATION:DEFINE1?  
-> :CURSOR:HORIZONTAL:CALCULATION:  
DEFINE1 "V1"

**:CURSOR:HORIZONTAL:CALCULATION:STATE<x>**

Function Turns ON/OFF the calculation item of the horizontal cursor or queries the current setting.

Syntax :CURSOR:HORIZONTAL:CALCULATION:STATE<x>  
{<Boolean>}  
:CURSOR:HORIZONTAL:CALCULATION:  
STATE<x>?  
<x> = 1 to 4

Example :CURSOR:HORIZONTAL:CALCULATION:STATE1 ON  
:CURSOR:HORIZONTAL:CALCULATION:STATE1?  
-> :CURSOR:HORIZONTAL:CALCULATION:  
STATE1 1

**:CURSOR:HORIZONTAL:CALCULATION:VALUE<x>?**

Function Queries the measured value of the calculation item of the horizontal cursor.

Syntax :CURSOR:HORIZONTAL:CALCULATION:  
VALUE<x>?  
<x> = 1 to 4

Example :CURSOR:HORIZONTAL:CALCULATION:VALUE1?  
-> :CURSOR:HORIZONTAL:CALCULATION:  
VALUE1 0.000E+00

## 5.9 CURSOR Group

### **:CURSOR:HORIZONTAL:TRACE**

**Function** Sets the source trace of the horizontal cursor or queries the current setting.

**Syntax** :CURSOR:HORIZONTAL:TRACE {<NRf>  
:CURSOR:HORIZONTAL:TRACE?  
<NRf> = 1 to 8

**Example** :CURSOR:HORIZONTAL:TRACE 1  
:CURSOR:HORIZONTAL:TRACE? -> :CURSOR:  
HORIZONTAL:TRACE 1

### **:CURSOR:MARKER?**

**Function** Queries all settings related to the marker cursors.

**Syntax** :CURSOR:MARKER?

**Example** :CURSOR:MARKER? -> :CURSOR:MARKER:  
CALCULATION:DEFINE1 "T2";DEFINE2 "V2";  
DEFINE3 "V3";DEFINE4 "V4";STATE1 0;  
STATE2 0;STATE3 0;STATE4 0;:CURSOR:  
MARKER:CM1:DT2:STATE 1;:CURSOR:MARKER:  
CM1:DT3:STATE 1;:CURSOR:MARKER:CM1:DT4:  
STATE 1;:CURSOR:MARKER:CM1:DV2:  
STATE 1;:CURSOR:MARKER:CM1:DV3:  
STATE 1;:CURSOR:MARKER:CM1:DV4:  
STATE 1;:CURSOR:MARKER:CM1:  
POSITION 1.000E+00;T:STATE 1;:CURSOR:  
MARKER:CM1:TRACE 1;V:STATE 1;:CURSOR:  
MARKER:CM2:DT1:STATE 1;:CURSOR:MARKER:  
CM2:DT3:STATE 1;:CURSOR:MARKER:CM2:DT4:  
STATE 1;:CURSOR:MARKER:CM2:DV1:  
STATE 1;:CURSOR:MARKER:CM2:DV3:  
STATE 1;:CURSOR:MARKER:CM2:DV4:  
STATE 1;:CURSOR:MARKER:CM2:  
POSITION 1.000E+00;T:STATE 1;:CURSOR:  
MARKER:CM2:TRACE 1;V:STATE 1;:CURSOR:  
MARKER:CM3:DT1:STATE 1;:CURSOR:MARKER:  
CM3:DT2:STATE 1;:CURSOR:MARKER:CM3:DT4:  
STATE 1;:CURSOR:MARKER:CM3:DV1:  
STATE 1;:CURSOR:MARKER:CM3:DV2:  
STATE .....

### **:CURSOR:MARKER:CALCULATION?**

**Function** Queries all settings related to calculation items of the marker cursors.

**Syntax** :CURSOR:MARKER:CALCULATION?

**Example** :CURSOR:MARKER:CALCULATION? -> :CURSOR:  
MARKER:CALCULATION:  
DEFINE1 "T2";DEFINE2 "V2";DEFINE3 "V3";  
DEFINE4 "V4";STATE1 0;STATE2 0;  
STATE3 0;STATE4 0

### **:CURSOR:MARKER:CALCULATION:ALL**

**Function** Turns ON/OFF all calculation items of the marker cursors.

**Syntax** :CURSOR:MARKER:CALCULATION:ALL  
{<Boolean>}

**Example** :CURSOR:MARKER:CALCULATION:ALL ON

### **:CURSOR:MARKER:CALCULATION:DEFINE<x>**

**Function** Sets the equation of the calculation item of the marker cursors or queries the current setting.

**Syntax** :CURSOR:MARKER:CALCULATION:DEFINE<x>  
{<String>}  
:CURSOR:MARKER:CALCULATION:DEFINE<x>?  
<x> = 1 to 4  
<String> = Up to 128 characters

**Example** :CURSOR:MARKER:CALCULATION:DEFINE1 "T1"  
:CURSOR:MARKER:CALCULATION:DEFINE1?  
-> :CURSOR:MARKER:CALCULATION:  
DEFINE1 "T1"

### **:CURSOR:MARKER:CALCULATION:STATE<x>**

**Function** Turns ON/OFF the calculation item of the marker cursors or queries the current setting.

**Syntax** :CURSOR:MARKER:CALCULATION:STATE<x>  
{<Boolean>}  
:CURSOR:MARKER:CALCULATION:STATE<x>?  
<x> = 1 to 4

**Example** :CURSOR:MARKER:CALCULATION:STATE1 ON  
:CURSOR:MARKER:CALCULATION:STATE1?  
-> :CURSOR:MARKER:CALCULATION:STATE1 1

### **:CURSOR:MARKER:CALCULATION:VALUE<x>?**

**Function** Queries the measured value of the calculation item of the marker cursors.

**Syntax** :CURSOR:MARKER:CALCULATION:VALUE<x>?  
<x> = 1 to 4

**Example** :CURSOR:MARKER:CALCULATION:VALUE1?  
-> :CURSOR:MARKER:CALCULATION:  
VALUE1 0.000E+00

### **:CURSOR:MARKER:CM<x>?**

**Function** Queries all settings related to the marker cursor.

**Syntax** :CURSOR:MARKER:CM<x>?  
<x> = 1 to 4

**Example** :CURSOR:MARKER:CM1? -> :CURSOR:MARKER:  
CM1:DT2:STATE 1;:CURSOR:  
MARKER:CM1:DT3:STATE 1;:CURSOR:MARKER:  
CM1:DT4:STATE 1;:CURSOR:MARKER:CM1:DV2:  
STATE 1;:CURSOR:MARKER:CM1:DV3:  
STATE 1;:CURSOR:MARKER:CM1:DV4:  
STATE 1;:CURSOR:MARKER:CM1:  
POSITION 1.000E+00;T:STATE 1;:CURSOR:  
MARKER:CM1:TRACE 1;V:STATE 1

**:CURSOR:MARKER:CM<x>:ALL**

Function Turns ON/OFF all items of the marker cursor.

Syntax :CURSOR:MARKER:CM<x>:ALL {<Boolean>}  
<x> = 1 to 4

Example :CURSOR:MARKER:CM1:ALL ON

**:CURSOR:MARKER:CM<x>:DT<x>?**

Function Queries all settings related to the  $\Delta T$  measurement of the cursor marker.

Syntax :CURSOR:MARKER:CM<x>:DT<x>?  
<x> of CM<x> = 1 to 4  
<x> of DT<x> = 1 to 4

Example :CURSOR:MARKER:CM1:DT1? -> :CURSOR:  
MARKER:CM1:DT1:STATE 1

**:CURSOR:MARKER:CM<x>:DT<x>:STATE**

Function Turns ON/OFF the  $\Delta T$  measurement of the maker cursor or queries the current setting.

Syntax :CURSOR:MARKER:CM<x>:DT<x>:STATE  
{<Boolean>}  
:CURSOR:MARKER:CM<x>:DT<x>:STATE?  
<x> of CM<x> = 1 to 4  
<x> of DT<x> = 1 to 4

Example :CURSOR:MARKER:CM1:DT2:STATE ON  
:CURSOR:MARKER:CM1:DT2:STATE?  
-> :CURSOR:MARKER:CM1:DT2:STATE 1

**:CURSOR:MARKER:CM<x>:DT<x>:VALUE?**

Function Queries the  $\Delta T$  value of the marker cursor.

Syntax :CURSOR:MARKER:CM<x>:DT<x>:VALUE?  
<x> of CM<x> = 1 to 4  
<x> of DT<x> = 1 to 4

Example :CURSOR:MARKER:CM1:DT2:VALUE?  
-> :CURSOR:MARKER:CM1:DT2:  
VALUE 0.000E+00

**:CURSOR:MARKER:CM<x>:DV<x>?**

Function Queries all settings related to the  $\Delta V$  measurement of the cursor marker.

Syntax :CURSOR:MARKER:CM<x>:DV<x>?  
<x> of CM<x> = 1 to 4  
<x> of DV<x> = 1 to 4

Example :CURSOR:MARKER:CM1:DV2? -> :CURSOR:  
MARKER:CM1:DV2:STATE 1

**:CURSOR:MARKER:CM<x>:DV<x>:STATE**

Function Turns ON/OFF the  $\Delta V$  measurement of the maker cursor or queries the current setting.

Syntax :CURSOR:MARKER:CM<x>:DV<x>:STATE  
{<Boolean>}  
:CURSOR:MARKER:CM<x>:DV<x>:STATE?  
<x> of CM<x> = 1 to 4  
<x> of DV<x> = 1 to 4

Example :CURSOR:MARKER:CM1:DV2:STATE ON  
:CURSOR:MARKER:CM1:DV2:STATE?  
-> :CURSOR:MARKER:CM1:DV2:STATE 1

**:CURSOR:MARKER:CM<x>:DV<x>:VALUE?**

Function Queries the  $\Delta V$  value of the marker cursor.

Syntax :CURSOR:MARKER:CM<x>:DV<x>:VALUE?  
<x> of CM<x> = 1 to 4  
<x> of DV<x> = 1 to 4

Example :CURSOR:MARKER:CM1:DV2:VALUE?  
-> :CURSOR:MARKER:CM1:DV2:VALUE 0

**:CURSOR:MARKER:CM<x>:JUMP**

Function Jumps the marker cursor to the center position of the zoom waveform.

Syntax :CURSOR:MARKER:CM<x>:JUMP {Z1|Z2}  
<x> = 1 to 4

Example :CURSOR:MARKER:CM1:JUMP Z1

**:CURSOR:MARKER:CM<x>:POSITION**

Function Sets the marker cursor position or queries the current setting.

Syntax :CURSOR:MARKER:CM<x>:POSITION {<Nrf>}  
:CURSOR:MARKER:CM<x>:POSITION?  
<x> = 1 to 4  
<Nrf> = -5 to 5 div

Example :CURSOR:MARKER:CM1:POSITION 1  
:CURSOR:MARKER:CM1:POSITION?  
-> :CURSOR:MARKER:CM1:POSITION 1.000E+00

**:CURSOR:MARKER:CM<x>:T?**

Function Queries all settings related to the time measurement of the marker cursor.

Syntax :CURSOR:MARKER:CM<x>:T?  
<x> = 1 to 4

Example :CURSOR:MARKER:CM1:T? -> :CURSOR:  
MARKER:CM1:T:STATE 1

## 5.9 CURSOR Group

### **:CURSOR:MARKER:CM<x>:T:STATE**

**Function** Turns ON/OFF the time measurement of the maker cursor or queries the current setting.

**Syntax** :CURSOR:MARKER:CM<x>:T:STATE  
{<Boolean>}  
:CURSOR:MARKER:CM<x>:T:STATE?  
<x> = 1 to 4

**Example** :CURSOR:MARKER:CM1:T:STATE ON  
:CURSOR:MARKER:CM1:T:STATE? -> :CURSOR:  
MARKER:CM1:T:STATE 1

### **:CURSOR:MARKER:CM<x>:T:VALUE?**

**Function** Queries the time value at the marker cursor position.

**Syntax** :CURSOR:MARKER:CM<x>:T:VALUE?  
<x> = 1 to 4

**Example** :CURSOR:MARKER:CM1:T:VALUE? -> :CURSOR:  
MARKER:CM1:T:VALUE 0.000E+00

### **:CURSOR:MARKER:CM<x>:TRACE**

**Function** Sets the source trace of the marker cursor or queries the current setting.

**Syntax** :CURSOR:MARKER:CM<x>:TRACE {<NRf>|OFF}  
:CURSOR:MARKER:CM<x>:TRACE?  
<x> = 1 to 4  
<NRf> = 1 to 8

**Example** :CURSOR:MARKER:CM1:TRACE 1  
:CURSOR:MARKER:CM1:TRACE? -> :CURSOR:  
MARKER:CM1:TRACE 1

### **:CURSOR:MARKER:CM<x>:V?**

**Function** Queries all settings related to the voltage measurement of the marker cursor.

**Syntax** :CURSOR:MARKER:CM<x>:V?  
<x> = 1 to 4

**Example** :CURSOR:MARKER:CM1:V? -> :CURSOR:  
MARKER:CM1:V:STATE 1

### **:CURSOR:MARKER:CM<x>:V:STATE**

**Function** Turns ON/OFF the voltage measurement of the maker cursor or queries the current setting.

**Syntax** :CURSOR:MARKER:CM<x>:V:STATE  
{<Boolean>}  
:CURSOR:MARKER:CM<x>:V:STATE?  
<x> = 1 to 4

**Example** :CURSOR:MARKER:CM1:V:STATE ON  
:CURSOR:MARKER:CM1:V:STATE? -> :CURSOR:  
MARKER:CM1:V:STATE 1

### **:CURSOR:MARKER:CM<x>:V:VALUE?**

**Function** Queries the voltage value at the marker cursor position.

**Syntax** :CURSOR:MARKER:CM<x>:V:VALUE?  
<x> = 1 to 4

**Example** :CURSOR:MARKER:CM1:V:VALUE? -> :CURSOR:  
MARKER:CM1:V:VALUE 0.000E+00

### **:CURSOR:MARKER:FORM**

**Function** Sets the marker cursor form or queries the current setting.

**Syntax** :CURSOR:MARKER:FORM {LINE|MARK}  
:CURSOR:MARKER:FORM?

**Example** :CURSOR:MARKER:FORM LINE  
:CURSOR:MARKER:FORM?  
-> :CURSOR:MARKER:FORM LINE

### **:CURSOR:SERIAL?**

**Function** Queries all settings related to the serial cursors.

**Syntax** :CURSOR:SERIAL?

**Example** :CURSOR:SERIAL? -> :CURSOR:SERIAL:  
SCURSOR1:ACTIVE HIGH;  
BCOUNT 8;BITRATE 10.00E+00;  
BITORDER MSBFIRST;FORMAT BINARY;  
HYSTERESIS 1.000E+00;LEVEL 0.000E+00;  
MODE 1;POSITION -4.000E+00;TRACE 1;  
TRACK OFF;:CURSOR:SERIAL:SCURSOR2:  
ACTIVE HIGH;BCOUNT 8;BITRATE 1.000E+00;  
BITORDER MSBFIRST;FORMAT BINARY;  
HYSTERESIS 3.000E+00;LEVEL 0.000E+00;  
MODE 0;POSITION -4.000E+00;TRACE 2;  
TRACK OFF

### **:CURSOR:SERIAL:SCURSOR<x>?**

**Function** Queries all settings related to the serial cursor.

**Syntax** :CURSOR:SERIAL:SCURSOR<x>?  
<x> = 1 or 2

**Example** :CURSOR:SERIAL:SCURSOR1? -> :CURSOR:  
SERIAL:SCURSOR1:ACTIVE HIGH;  
BCOUNT 8;BITRATE 1.25E+06;  
BITORDER MSBFIRST;FORMAT BINARY;  
HYSTERESIS 300.00E-03;LEVEL 0.000E+00;  
MODE 1;POSITION -4.00E+00;TRACE 1;  
TRACK OFF

### **:CURSOR:SERIAL:SCURSOR<x>:ACTIVE**

**Function** Sets the active level of the serial cursor or queries the current setting.

**Syntax** :CURSOR:SERIAL:SCURSOR<x>:ACTIVE {HIGH|  
LOW}  
:CURSOR:SERIAL:SCURSOR<x>:ACTIVE?  
<x> = 1 or 2

**Example** :CURSOR:SERIAL:SCURSOR1:ACTIVE HIGH  
:CURSOR:SERIAL:SCURSOR1:ACTIVE?  
-> :CURSOR:SERIAL:SCURSOR1:ACTIVE HIGH

**:CURSor:SErIal:SCURsor<x>:BCOunt**

Function Sets the bit length of the serial cursor or queries the current setting.

Syntax :CURSor:SErIal:SCURsor<x>:BCOunt  
{<NRf>}  
:CURSor:SErIal:SCURsor<x>:BCOunt?  
<x> = 1 or 2  
<NRf> = 1 to 128 (bits)

Example :CURSOR:SERIAL:SCURSOR1:BCOUNT 8  
:CURSOR:SERIAL:SCURSOR1:BCOUNT?  
-> :CURSOR:SERIAL:SCURSOR1:BCOUNT 8

**:CURSor:SErIal:SCURsor<x>:BITRate**

Function Sets the bit rate of the serial cursor or queries the current setting.

Syntax :CURSor:SErIal:SCURsor<x>:BITRate  
{<NRf>}  
:CURSor:SErIal:SCURsor<x>:BITRate?  
<x> = 1 or 2  
<NRf> = 1 to 1 G (bps)

Example :CURSOR:SERIAL:SCURSOR1:BITRATE 10  
:CURSOR:SERIAL:SCURSOR1:BITRATE?  
-> :CURSOR:SERIAL:SCURSOR1:  
BITRATE 10.00E+00

**:CURSor:SErIal:SCURsor<x>:BITOrder**

Function Sets the bit order of the serial cursor or queries the current setting.

Syntax :CURSor:SErIal:SCURsor<x>:BITOrder  
{LSBFirst|MSBFirst}  
:CURSor:SErIal:SCURsor<x>:BITOrder?  
<x> = 1 or 2

Example :CURSOR:SERIAL:SCURSOR1:  
BITORDER LSBFIRST  
:CURSOR:SERIAL:SCURSOR1:BITORDER?  
-> :CURSOR:SERIAL:SCURSOR1:  
BITORDER LSBFIRST

**:CURSor:SErIal:SCURsor<x>:FORMat**

Function Sets the display format of the serial cursor or queries the current setting.

Syntax :CURSor:SErIal:SCURsor<x>:FORMat  
{BINary|HEXa}  
:CURSor:SErIal:SCURsor<x>:FORMat?  
<x> = 1 or 2

Example :CURSOR:SERIAL:SCURSOR1:FORMAT HEXA  
:CURSOR:SERIAL:SCURSOR1:FORMAT?  
-> :CURSOR:SERIAL:SCURSOR1:FORMAT HEXA

**:CURSor:SErIal:SCURsor<x>:HYSTeresis**

Function Sets the hysteresis of the serial cursor or queries the current setting.

Syntax :CURSor:SErIal:SCURsor<x>:HYSTeresis  
{<NRf>}  
:CURSor:SErIal:SCURsor<x>:HYSTeresis?  
<x> = 1 or 2  
<NRf> = 0 to 4 (div)

Example :CURSOR:SERIAL:SCURSOR1:HYSTERESIS 1  
:CURSOR:SERIAL:SCURSOR1:HYSTERESIS?  
-> :CURSOR:SERIAL:SCURSOR1:  
HYSTERESIS 1.000E+00

**:CURSor:SErIal:SCURsor<x>:JUMP**

Function Moves the serial cursor to the specified direction.

Syntax :CURSor:SErIal:SCURsor<x>:JUMP  
{BACK|FRONT}  
<x> = 1 or 2

Example :CURSOR:SERIAL:SCURSOR1:JUMP BACK

**:CURSor:SErIal:SCURsor<x>:LEVel**

Function Sets the threshold level of the serial cursor or queries the current setting.

Syntax :CURSor:SErIal:SCURsor<x>:LEVel  
{<Voltage>|<Current>}  
:CURSor:SErIal:SCURsor<x>:LEVel?  
<x> = 1 or 2  
<Voltage> and <Current> = See the DL9000 User's Manual.

Example :CURSOR:SERIAL:SCURSOR1:LEVEL 0V  
:CURSOR:SERIAL:SCURSOR1:LEVEL?  
-> :CURSOR:SERIAL:SCURSOR1:  
LEVEL 0.000E+00

**:CURSor:SErIal:SCURsor<x>:MODE**

Function Turns ON/OFF the serial cursor or queries the current setting.

Syntax :CURSor:SErIal:SCURsor<x>:MODE  
{<Boolean>}  
:CURSor:SErIal:SCURsor<x>:MODE?  
<x> = 1 or 2

Example :CURSOR:SERIAL:SCURSOR1:MODE ON  
:CURSOR:SERIAL:SCURSOR1:MODE?  
-> :CURSOR:SERIAL:SCURSOR1:MODE 1

**:CURSor:SErIal:SCURsor<x>:POSition**

Function Sets the serial cursor position or queries the current setting.

Syntax :CURSor:SErIal:SCURsor<x>:POSition  
{<NRf>}  
:CURSor:SErIal:SCURsor<x>:POSition?  
<x> = 1 or 2  
<NRf> = -5 to 5 div

Example :CURSOR:SERIAL:SCURSOR1:POSITION 1  
:CURSOR:SERIAL:SCURSOR1:POSITION?  
-> :CURSOR:SERIAL:SCURSOR1:  
POSITION 1.000E+00

## 5.9 CURSOR Group

### **:CURSOR:SERIAL:SCURSOR<x>:TRACE**

**Function** Sets the trace of the serial cursor or queries the current setting.

**Syntax** :CURSOR:SERIAL:SCURSOR<x>:TRACE {<NRF>}  
:CURSOR:SERIAL:SCURSOR<x>:TRACE?  
<x> = 1 or 2  
<NRF> = 1 to 8

**Example** :CURSOR:SERIAL:SCURSOR1:TRACE 1  
:CURSOR:SERIAL:SCURSOR1:TRACE?  
-> :CURSOR:SERIAL:SCURSOR1:TRACE 1

### **:CURSOR:SERIAL:SCURSOR<x>:TRACK**

**Function** Jumps the serial cursor onto the zoom waveform.

**Syntax** :CURSOR:SERIAL:SCURSOR<x>:TRACK {OFF|Z1|Z2}  
:CURSOR:SERIAL:SCURSOR<x>:TRACK?  
<x> = 1 or 2

**Example** :CURSOR:SERIAL:SCURSOR1:TRACK OFF  
:CURSOR:SERIAL:SCURSOR1:TRACK?  
-> :CURSOR:SERIAL:SCURSOR1:TRACK OFF

### **:CURSOR:SERIAL:SCURSOR<x>:VALUE?**

**Function** Queries the measured value of the serial cursor.

**Syntax** :CURSOR:SERIAL:SCURSOR<x>:VALUE?  
<x> = 1 or 2

**Example** :CURSOR:SERIAL:SCURSOR1:VALUE?  
-> :CURSOR:SERIAL:SCURSOR1:  
VALUE "11000111"

### **:CURSOR:TYPE**

**Function** Sets the cursor type or queries the current setting.

**Syntax** :CURSOR:TYPE {HORIZONTAL|HAVERTICAL|  
MARKER|SERIAL|VERTICAL|VT}  
:CURSOR:TYPE?

**Example** :CURSOR:TYPE HORIZONTAL  
:CURSOR:TYPE?  
-> :CURSOR:TYPE HORIZONTAL

### **:CURSOR:VERTICAL?**

**Function** Queries all settings related to the vertical cursors.

**Syntax** :CURSOR:VERTICAL?

**Example** :CURSOR:VERTICAL? -> :CURSOR:VERTICAL:  
BASIC:DT:STATE 1;:CURSOR:VERTICAL:  
BASIC:LINKAGE 1;PERDT:STATE 1;:CURSOR:  
VERTICAL:BASIC:T1:POSITION -4.000E+00;  
STATE 1;:CURSOR:VERTICAL:BASIC:T2:  
POSITION -4.000E+00;STATE 1;:CURSOR:  
VERTICAL:CALCULATION:DEFINE1 "T1";  
DEFINE2 "T2";DEFINE3 "T1";DEFINE4 "T2";  
STATE1 0;STATE2 0;STATE3 0;STATE4 0

### **:CURSOR:VERTICAL:BASIC?**

**Function** Queries all settings related to basic items of the vertical cursors.

**Syntax** :CURSOR:VERTICAL:BASIC?

**Example** :CURSOR:VERTICAL:BASIC? -> :CURSOR:  
VERTICAL:BASIC:DT:STATE 1;:  
CURSOR:VERTICAL:BASIC:LINKAGE 1;PERDT:  
STATE 1;:CURSOR:VERTICAL:BASIC:T1:  
POSITION -4.000E+00;STATE 1;:CURSOR:  
VERTICAL:BASIC:T2:POSITION -4.000E+00;  
STATE 1

### **:CURSOR:VERTICAL[:BASIC]:ALL**

**Function** Turns ON/OFF all basic items of the vertical cursors.

**Syntax** :CURSOR:VERTICAL[:BASIC]:ALL  
{<Boolean>}

**Example** :CURSOR:VERTICAL:BASIC:ALL ON

### **:CURSOR:VERTICAL[:BASIC]:DT?**

**Function** Queries all settings related to the  $\Delta T$  measurement of the vertical cursors.

**Syntax** :CURSOR:VERTICAL[:BASIC]:DT?

**Example** :CURSOR:VERTICAL:BASIC:DT? -> :CURSOR:  
VERTICAL:BASIC:DT:STATE 1

### **:CURSOR:VERTICAL[:BASIC]:DT:STATE**

**Function** Turns ON/OFF the  $\Delta T$  measurement of the vertical cursors or queries the current setting.

**Syntax** :CURSOR:VERTICAL[:BASIC]:DT:STATE  
{<Boolean>}  
:CURSOR:VERTICAL[:BASIC]:DT:STATE?

**Example** :CURSOR:VERTICAL:BASIC:DT:STATE ON  
:CURSOR:VERTICAL:BASIC:DT:STATE?  
-> :CURSOR:VERTICAL:BASIC:DT:STATE 1

### **:CURSOR:VERTICAL[:BASIC]:DT:VALUE?**

**Function** Queries the  $\Delta T$  value of the vertical cursors.

**Syntax** :CURSOR:VERTICAL[:BASIC]:DT:VALUE?

**Example** :CURSOR:VERTICAL:BASIC:DT:VALUE?  
-> :CURSOR:VERTICAL:BASIC:DT:  
VALUE 1.000E+00

### **:CURSOR:VERTICAL[:BASIC]:LINKAGE**

**Function** Turns ON/OFF the vertical cursor link or queries the current setting.

**Syntax** :CURSOR:VERTICAL[:BASIC]:  
LINKAGE {<Boolean>}

:CURSOR:VERTICAL[:BASIC]:LINKAGE?

**Example** :CURSOR:VERTICAL:BASIC:LINKAGE ON  
:CURSOR:VERTICAL:BASIC:LINKAGE?  
-> :CURSOR:VERTICAL:BASIC:LINKAGE 1



**:CURSOR:VERTICAL[:BASIC]:PERDt?**

**Function** Queries all settings related to the  $1/\Delta T$  measurement of the vertical cursors.

**Syntax** :CURSOR:VERTICAL[:BASIC]:PERDt?

**Example** :CURSOR:VERTICAL:BASIC:PERDt?  
-> :CURSOR:VERTICAL:BASIC:PERDt:STATE 1

**:CURSOR:VERTICAL[:BASIC]:PERDt:STATE**

**Function** Turns ON/OFF the  $1/\Delta T$  measurement of the vertical cursors or queries the current setting.

**Syntax** :CURSOR:VERTICAL[:BASIC]:PERDt:STATE  
{<Boolean>}  
:CURSOR:VERTICAL[:BASIC]:PERDt:STATE?

**Example** :CURSOR:VERTICAL:BASIC:PERDt:STATE ON  
:CURSOR:VERTICAL:BASIC:PERDt:STATE?  
-> :CURSOR:VERTICAL:BASIC:PERDt:STATE 1

**:CURSOR:VERTICAL[:BASIC]:PERDt:VALUE?**

**Function** Queries the  $1/\Delta T$  value of the vertical cursors.

**Syntax** CURSOR:VERTICAL[:BASIC]:PERDt:VALUE?

**Example** CURSOR:VERTICAL:BASIC:PERDt:VALUE?  
-> :CURSOR:VERTICAL:BASIC:PERDt:  
VALUE 0.000E+00

**:CURSOR:VERTICAL[:BASIC]:T<x>?**

**Function** Queries all settings related to the vertical cursor.

**Syntax** :CURSOR:VERTICAL[:BASIC]:T<x>?  
<x> = 1 or 2

**Example** :CURSOR:VERTICAL:BASIC:T1? -> :CURSOR:  
VERTICAL:BASIC:T1:  
POSITION -4.000E+00;STATE 1

**:CURSOR:VERTICAL[:BASIC]:T<x>:JUMP**

**Function** Jumps to the center position of the zoom waveform of the vertical cursor.

**Syntax** :CURSOR:VERTICAL[:BASIC]:T<x>:JUMP {Z1|  
Z2}  
<x> = 1 or 2

**Example** :CURSOR:VERTICAL:BASIC:T1:JUMP Z1

**:CURSOR:VERTICAL[:BASIC]:T<x>:POSITION**

**Function** Sets the vertical cursor position or queries the current setting.

**Syntax** :CURSOR:VERTICAL[:BASIC]:T<x>:POSITION  
{<NRf>}  
:CURSOR:VERTICAL[:BASIC]:T<x>:POSITION?  
<x> = 1 or 2  
<NRf> = -5 to 5 (div)

**Example** :CURSOR:VERTICAL:BASIC:T1:POSITION 1  
:CURSOR:VERTICAL:BASIC:T1:POSITION?  
-> :CURSOR:VERTICAL:BASIC:T1:  
POSITION 1.000E+00

**:CURSOR:VERTICAL[:BASIC]:T<x>:STATE**

**Function** Turns ON/OFF the vertical cursor or queries the current setting.

**Syntax** :CURSOR:VERTICAL[:BASIC]:T<x>:STATE  
{<Boolean>}  
:CURSOR:VERTICAL[:BASIC]:T<x>:STATE?  
<x> = 1 or 2

**Example** :CURSOR:VERTICAL:BASIC:T1:STATE ON  
:CURSOR:VERTICAL:BASIC:T1:STATE?  
-> :CURSOR:VERTICAL:BASIC:T1:STATE 1

**:CURSOR:VERTICAL[:BASIC]:T<x>:VALUE?**

**Function** Queries the time value at the vertical cursor.

**Syntax** :CURSOR:VERTICAL[:BASIC]:T<x>:VALUE?  
<x> = 1 or 2

**Example** :CURSOR:VERTICAL:BASIC:T1:VALUE?  
-> :CURSOR:VERTICAL:BASIC:T1:  
VALUE 0.000E+00

**:CURSOR:VERTICAL:CALCULATION?**

**Function** Queries all settings related to calculation items of the vertical cursors.

**Syntax** :CURSOR:VERTICAL:CALCULATION?

**Example** :CURSOR:VERTICAL:CALCULATION?  
-> :CURSOR:VERTICAL:CALCULATION:  
DEFINE1 "T1";DEFINE2 "T2";DEFINE3 "T1";  
DEFINE4 "T2";STATE1 0;STATE2 0;  
STATE3 0;STATE4 0

**:CURSOR:VERTICAL:CALCULATION:ALL**

**Function** Turns ON/OFF all calculation items of the vertical cursors.

**Syntax** :CURSOR:VERTICAL:CALCULATION:ALL  
{<Boolean>}

**Example** :CURSOR:VERTICAL:CALCULATION:ALL ON

## 5.9 CURSOR Group

### **:CURSOR:VERTICAL:CALCULATION:**

#### **DEFine<x>**

**Function** Sets the equation of the calculation item of the vertical cursor or queries the current setting.

**Syntax** :CURSOR:VERTICAL:CALCULATION:DEFine<x>  
{<String>}  
:CURSOR:VERTICAL:CALCULATION:DEFine<x>?  
<x> = 1 to 4  
<String> = Up to 128 characters

**Example** :CURSOR:VERTICAL:CALCULATION:  
DEFINE1 "T1"  
:CURSOR:VERTICAL:CALCULATION:DEFINE1?  
-> :CURSOR:VERTICAL:CALCULATION:  
DEFINE1 "T1"

### **:CURSOR:VERTICAL:CALCULATION:**

#### **STate<x>**

**Function** Turns ON/OFF the calculation item of the vertical cursor or queries the current setting.

**Syntax** :CURSOR:VERTICAL:CALCULATION:STate<x>  
{<Boolean>}  
:CURSOR:VERTICAL:CALCULATION:STate<x>?  
<x> = 1 to 4

**Example** :CURSOR:VERTICAL:CALCULATION:STATE1 ON  
:CURSOR:VERTICAL:CALCULATION:STATE1?  
-> :CURSOR:VERTICAL:CALCULATION:  
STATE1 1

### **:CURSOR:VERTICAL:CALCULATION:**

#### **VALue<x>?**

**Function** Queries the measured value of the calculation item of the vertical cursor.

**Syntax** :CURSOR:VERTICAL:CALCULATION:VALue<x>?  
<x> = 1 to 4

**Example** :CURSOR:VERTICAL:CALCULATION:VALUE1?  
-> :CURSOR:VERTICAL:CALCULATION:  
VALUE1 0.000E+00

### **:CURSOR:VT?**

**Function** Queries all settings related to the VT cursor.

**Syntax** :CURSOR:VT?

**Example** :CURSOR:VT? -> :CURSOR:VT: BASIC:  
POSITION 0.000E+00;T:  
STATE 1;:CURSOR:VT: BASIC:V1:STATE 1;:  
CURSOR:VT: BASIC:V2:STATE 1;:CURSOR:VT:  
BASIC:V3:STATE 1;:CURSOR:VT: BASIC:V4:  
STATE 1;:CURSOR:VT: BASIC:V5:STATE 1;:  
CURSOR:VT: BASIC:V6:STATE 1;:CURSOR:VT:  
BASIC:V7:STATE 1;:CURSOR:VT: BASIC:V8:  
STATE 1;:CURSOR:VT: CALCULATION:  
DEFINE1 "T1";DEFINE2 "V(C1)";  
DEFINE3 "V(C2)";DEFINE4 "V(C3)";  
STATE1 0;STATE2 0;STATE3 0;STATE4 0

### **:CURSOR:VT:BASic?**

**Function** Queries all settings related to basic items of the VT cursor.

**Syntax** :CURSOR:VT:BASic?

**Example** :CURSOR:VT:BASic? -> :CURSOR:VT: BASIC:  
POSITION 0.000E+00;T:  
STATE 1;:CURSOR:VT: BASIC:V1:STATE 1;:  
CURSOR:VT: BASIC:V2:STATE 1;:CURSOR:VT:  
BASIC:V3:STATE 1;:CURSOR:VT: BASIC:V4:  
STATE 1;:CURSOR:VT: BASIC:V5:STATE 1;:  
CURSOR:VT: BASIC:V6:STATE 1;:CURSOR:VT:  
BASIC:V7:STATE 1;:CURSOR:VT: BASIC:V8:  
STATE 1

### **:CURSOR:VT[:BASic]:ALL**

**Function** Turns ON/OFF all basic items of the VT cursor.

**Syntax** :CURSOR:VT[:BASic]:ALL {<Boolean>}

**Example** :CURSOR:VT: BASIC:ALL ON

### **:CURSOR:VT[:BASic]:POSition**

**Function** Sets the VT cursor position or queries the current setting.

**Syntax** :CURSOR:VT[:BASic]:POSition {<Nrf>}  
:CURSOR:VT[:BASic]:POSition?  
<NRF> = -5 to 5 (div)

**Example** :CURSOR:VT: BASIC:POSITION 1  
:CURSOR:VT: BASIC:POSITION? -> :CURSOR:  
VT: BASIC:POSITION 1.000E+00

### **:CURSOR:VT[:BASic]:T?**

**Function** Queries all settings related to the time value of the VT cursor.

**Syntax** :CURSOR:VT[:BASic]:T?

**Example** :CURSOR:VT: BASIC:T? -> :CURSOR:VT:  
BASIC:T:STATE 1

### **:CURSOR:VT[:BASic]:T:STate**

**Function** Turns ON/OFF the time value of the VT cursor or queries the current setting.

**Syntax** :CURSOR:VT[:BASic]:T:STATE {<Boolean>}  
:CURSOR:VT[:BASic]:T:STATE?

**Example** :CURSOR:VT: BASIC:T:STATE ON  
:CURSOR:VT: BASIC:T:STATE? -> :CURSOR:  
VT: BASIC:T:STATE 1

### **:CURSOR:VT[:BASic]:T:VALue?**

**Function** Queries the time value at the VT cursor.

**Syntax** :CURSOR:VT[:BASic]:T:VALue?

**Example** :CURSOR:VT: BASIC:T:VALUE? -> :CURSOR:  
VT: BASIC:T:VALUE 0.000E+00

**:CURSOR:VT[:BASIC]:V<x>?**

Function Queries all settings related to the voltage value of the VT cursor.

Syntax :CURSOR:VT[:BASIC]:V<x>?  
<x> = 1 to 8

Example :CURSOR:VT:BASIC:V1? -> :CURSOR:VT:  
BASIC:V1:STATE 1

**:CURSOR:VT[:BASIC]:V<x>:STATE**

Function Turns ON/OFF the voltage value of the VT cursor or queries the current setting.

Syntax :CURSOR:VT[:BASIC]:V<x>:STATE  
{<Boolean>}  
:CURSOR:VT[:BASIC]:V<x>:STATE?  
<x> = 1 to 8

Example :CURSOR:VT:BASIC:V1:STATE ON  
:CURSOR:VT:BASIC:V1:STATE? -> :CURSOR:  
VT:BASIC:V1:STATE 1

**:CURSOR:VT[:BASIC]:V<x>:VALUE?**

Function Queries the voltage value at the VT cursor.

Syntax :CURSOR:VT[:BASIC]:V<x>:VALUE?  
<x> = 1 to 8

Example :CURSOR:VT:BASIC:V1:VALUE? -> :CURSOR:  
VT:BASIC:V1:VALUE 0.000E+00

**:CURSOR:VT:CALCULATION?**

Function Queries all settings related to calculation items of the VT cursor.

Syntax :CURSOR:VT:CALCULATION?

Example :CURSOR:VT:CALCULATION? -> :CURSOR:VT:  
CALCULATION:DEFINE1 "T1";  
DEFINE2 "V(C1)";DEFINE3 "V(C2)";  
DEFINE4 "V(C3)";STATE1 0;STATE2 0;  
STATE3 0;STATE4 0

**:CURSOR:VT:CALCULATION:ALL**

Function Turns ON/OFF all calculation items of the VT cursor.

Syntax :CURSOR:VT:CALCULATION:ALL {<Boolean>}

Example :CURSOR:VT:CALCULATION:ALL ON

**:CURSOR:VT:CALCULATION:DEFINE<x>**

Function Sets the equation of the calculation item of the VT cursor or queries the current setting.

Syntax :CURSOR:VT:CALCULATION:DEFINE<x>  
{<String>}  
:CURSOR:VT:CALCULATION:DEFINE<x>?  
<x> = 1 to 4  
<String> = Up to 128 characters

Example :CURSOR:VT:CALCULATION:DEFINE1 "T1"  
:CURSOR:VT:CALCULATION:DEFINE1?  
-> :CURSOR:VT:CALCULATION:DEFINE1 "T1"

**:CURSOR:VT:CALCULATION:STATE<x>**

Function Turns ON/OFF the calculation item of the VT cursor or queries the current setting.

Syntax :CURSOR:VT:CALCULATION:STATE<x>  
{<Boolean>}  
:CURSOR:VT:CALCULATION:STATE<x>?  
<x> = 1 to 4

Example :CURSOR:VT:CALCULATION:STATE1 ON  
:CURSOR:VT:CALCULATION:STATE1?  
-> :CURSOR:VT:CALCULATION:STATE1 0

**:CURSOR:VT:CALCULATION:VALUE<x>?**

Function Queries the measured value of the calculation item of the VT cursor.

Syntax :CURSOR:VT:CALCULATION:VALUE<x>?  
<x> = 1 to 4

Example :CURSOR:VT:CALCULATION:VALUE1?  
-> :CURSOR:VT:CALCULATION:  
VALUE1 0.000E+00

**:CURSOR:VT:JUMP**

Function Jumps the VT cursor to the center position of the zoom waveform.

Syntax :CURSOR:VT:JUMP {Z1|Z2}

Example :CURSOR:VT:JUMP Z1

## 5.10 DISPlay Group

### **:DISPlay?**

**Function** Queries all settings related to the display.

**Syntax** :DISPlay?

**Example** :DISPLAY? -> :DISPLAY:ACCUMULATE:  
GRADE INTENSITY;MODE 0;PERSISTENCE:  
COUNT INFINITY;MODE COUNT;  
TIME 1.000E+00;:DISPLAY:BLIGHT:  
AUTOFF 0;BRIGHTNESS 1;LCD 1;TIMEOUT 1;  
:DISPLAY:COLOR:CHANNEL1 BLUE;  
CHANNEL2 BGREEN;CHANNEL3 CYAN;  
CHANNEL4 DBLUE;MATH1 GRAY;MATH2 GREEN;  
MATH3 LBLUE;MATH4 LGREEN;  
REFERENCE1 MAGENTA;REFERENCE2 MGREEN;  
REFERENCE3 ORANGE;REFERENCE4 PINK;  
:DISPLAY:FORMAT SINGLE;GRATICULE GRID;  
INTENSITY:WAVEFORM 10;CURSOR 10;  
GRID 10;MARKER 10;ZBOX 10;  
:DISPLAY:INTERPOLATE SINE;MAPPING:  
MODE AUTO;TRACE1 1;TRACE2 1;TRACE3 1;  
TRACE4 1;TRACE5 1;TRACE6 1;TRACE7 1;  
TRACE8 1;:DISPLAY:TRANSLUCENT 1

### **:DISPlay:ACCumulate?**

**Function** Queries all settings related to the accumulated display of waveforms.

**Syntax** :DISPlay:ACCumulate?

**Example** :DISPLAY:ACCUMULATE? -> :DISPLAY:  
ACCUMULATE:GRADE INTENSITY;  
MODE 0;PERSISTENCE:COUNT INFINITY;  
MODE COUNT;TIME 1.000E+00

### **:DISPlay:ACCumulate:GRADE**

**Function** Sets the accumulate mode or queries the current setting.

**Syntax** :DISPlay:ACCumulate:GRADE {COLOr|  
INTensity}

**Example** :DISPLAY:ACCUMULATE:GRADE INTENSITY  
:DISPLAY:ACCUMULATE:GRADE? -> :DISPLAY:  
ACCUMULATE:GRADE INTENSITY

### **:DISPlay:ACCumulate:MODE**

**Function** Turns ON/OFF the accumulate mode or queries the current setting.

**Syntax** :DISPlay:ACCumulate:MODE {<Boolean>}  
:DISPlay:ACCumulate:MODE?

**Example** :DISPLAY:ACCUMULATE:MODE ON  
:DISPLAY:ACCUMULATE:MODE? -> :DISPLAY:  
ACCUMULATE:MODE 1

### **:DISPlay:ACCumulate:PERSistence?**

**Function** Queries all settings related to persistence.

**Syntax** :DISPlay:ACCumulate:PERSistence?

**Example** :DISPLAY:ACCUMULATE:PERSISTENCE?  
-> :DISPLAY:ACCUMULATE:PERSISTENCE:  
COUNT INFINITY;MODE COUNT;  
TIME 1.000E+00

### **:DISPlay:ACCumulate:PERSistence:**

**COUNT**

**Function** Sets the persistence count or queries the current setting.

**Syntax** :DISPlay:ACCumulate:PERSistence:COUNT  
{<NRf>|INFINITY}  
:DISPlay:ACCumulate:PERSistence:COUNT?  
<NRf> = 1 to (the maximum number of history pages  
at the current record length)

**Example** :DISPLAY:ACCUMULATE:PERSISTENCE:  
COUNT INFINITY  
:DISPLAY:ACCUMULATE:PERSISTENCE:COUNT?  
-> :DISPLAY:ACCUMULATE:PERSISTENCE:  
COUNT INFINITY

### **:DISPlay:ACCumulate:PERSistence:MODE**

**Function** Sets the persistence mode or queries the current setting.

**Syntax** :DISPlay:ACCumulate:PERSistence:MODE  
{COUNT|TIME}  
:DISPlay:ACCumulate:PERSistence:MODE?

**Example** :DISPLAY:ACCUMULATE:PERSISTENCE:  
MODE COUNT  
:DISPLAY:ACCUMULATE:PERSISTENCE:MODE?  
-> :DISPLAY:ACCUMULATE:PERSISTENCE:  
MODE COUNT

### **:DISPlay:ACCumulate:PERSistence:TIME**

**Function** Sets the persistence time or queries the current setting.

**Syntax** :DISPlay:ACCumulate:PERSistence:TIME  
{<Time>|INFINITY}  
:DISPlay:ACCumulate:PERSistence:TIME?  
<Time> = 100 ms to 1 s in 100-ms steps, 1 s to 10 s  
in 200-ms steps, 10 s to 100 s in 2-s steps

**Example** :DISPLAY:ACCUMULATE:PERSISTENCE:TIME 1S  
:DISPLAY:ACCUMULATE:PERSISTENCE:TIME?  
-> :DISPLAY:ACCUMULATE:PERSISTENCE:  
TIME 1.000E+00

**:DISPlay:BLIGHT?**

Function Queries all settings related to the backlight.

Syntax :DISPlay:BLIGHT?

Example :DISPLAY:BLIGHT? -> :DISPLAY:BLIGHT:  
AUTOOFF 0;BRIGHTNESS 1;  
LCD 1;TIMEOUT 1

**:DISPlay:BLIGHT:AUTOoff**

Function Sets the function that automatically turns the backlight off or queries the current setting.

Syntax :DISPlay:BLIGHT:AUTOoff {<Boolean>}  
:DISPlay:BLIGHT:AUTOoff?

Example :DISPLAY:BLIGHT:AUTOOFF ON  
:DISPLAY:BLIGHT:AUTOOFF? -> :DISPLAY:  
BLIGHT:AUTOOFF 1

**:DISPlay:BLIGHT:BRIGhtness**

Function Sets the brightness of the backlight or queries the current setting.

Syntax :DISPlay:BLIGHT:BRIGhtness {<Nrf>}  
:DISPlay:BLIGHT:BRIGhtness?  
<Nrf> = 1 to 8

Example :DISPLAY:BLIGHT:BRIGHTNESS 1  
:DISPLAY:BLIGHT:BRIGHTNESS?  
-> :DISPLAY:BLIGHT:BRIGHTNESS 1

**:DISPlay:BLIGHT:LCD**

Function Turns ON/OFF the backlight or queries the current setting.

Syntax :DISPlay:BLIGHT:LCD {<Boolean>}  
:DISPlay:BLIGHT:LCD?

Example :DISPLAY:BLIGHT:LCD ON  
:DISPLAY:BLIGHT:LCD? -> :DISPLAY:  
BLIGHT:LCD 1

**:DISPlay:BLIGHT:TIMEout**

Function Sets the timeout of the backlight or queries the current setting.

Syntax :DISPlay:BLIGHT:TIMEout {<Nrf>}  
:DISPlay:BLIGHT:TIMEout?  
<Nrf> = 1 to 60 (minutes)

Example :DISPLAY:BLIGHT:TIMEOUT 60  
:DISPLAY:BLIGHT:TIMEOUT? -> :DISPLAY:  
BLIGHT:TIMEOUT 60

**:DISPlay:COLor?**

Function Queries all settings related to the waveform display color.

Syntax :DISPlay:COLor?

Example :DISPLAY:COLOR? -> :DISPLAY:COLOR:  
CHANNEL1 BLUE;  
CHANNEL2 BGREEN;CHANNEL3 CYAN;  
CHANNEL4 DBLUE;MATH1 GRAY;MATH2 GREEN;  
MATH3 LBLUE;MATH4 LGREEN;  
REFERENCE1 MAGENTA;REFERENCE2 MGREEN;  
REFERENCE3 ORANGE;REFERENCE4 PINK

**:DISPlay:COLor: {CHANnel<x>|MATH<x>|REFeRence<x>}**

Function Sets the color of the waveform or queries the current setting.

Syntax :DISPlay:COLor: {CHANnel<x>|MATH<x>|  
REFeRence<x>} {BLUE|BGRen|CYAN|DBLue|G  
RAY|GREn|  
LBLue|LGRen|MAGenta|MGRen|ORANge|  
PINK|PURPle|RED|SPINK|YELLow}  
:DISPlay:COLor: {CHANnel<x>|MATH<x>|  
REFeRence<x>}?  
<x> of CHANnel<x> = 1 to 4  
<x> of MATH<x> = 1 to 4  
<x> of REFeRence<x> = 1 to 4

Example :DISPLAY:COLOR:CHANNEL1 BLUE  
:DISPLAY:COLOR:CHANNEL1? -> :DISPLAY:  
COLOR:CHANNEL1 BLUE

**:DISPlay:FORMat**

Function Sets the display format or queries the current setting.

Syntax :DISPlay:FORMat {DUAL|QUAD|SINGle|  
TRIad}  
:DISPlay:FORMat?

Example :DISPLAY:FORMAT SINGLE  
:DISPLAY:FORMAT?  
-> :DISPLAY:FORMAT SINGLE

**:DISPlay:GRATicule**

Function Sets the graticule (grid) or queries the current setting.

Syntax :DISPlay:GRATicule {CROSShair|FRAMe|  
GRID|LINE}  
:DISPlay:GRATicule?

Example :DISPLAY:GRATICULE CROSSHAIR  
:DISPLAY:GRATICULE? -> :DISPLAY:  
GRATICULE CROSSHAIR

**:DISPlay:INTENSity?**

Function Queries all settings related to the intensity of the displayed items.

Syntax :DISPlay:INTENSity?  
:DISPlay:INTENSity? ->

Example :DISPLAY:INTENSITY? -> :DISPLAY:  
INTENSITY:WAVEFORM 10;  
CURSOR 10;GRID 10;MARKER 10

**:DISPlay:INTENSity[:WAVEform]**

Function Sets the intensity of the waveform or queries the current setting.

Syntax :DISPlay:INTENSity[:WAVEform] {<Nrf>}  
:DISPlay:INTENSity[:WAVEform]?  
<Nrf> = 1 to 20

Example :DISPLAY:INTENSITY:WAVEFORM 10  
:DISPLAY:INTENSITY:WAVEFORM?  
-> :DISPLAY:INTENSITY:WAVEFORM 10

## 5.10 DISPlay Group

### **:DISPlay:INTENsity:{CURSor|GRID|MARKer|ZBOX}**

**Function** Sets the intensity of the display item or queries the current setting.

**Syntax** :DISPlay:INTENsity:{CURSor|GRID|MARKer|ZBOX} {<NRf>}  
:DISPlay:INTENsity:{CURSor|GRID|MARKer|ZBOX}?  
<NRf> = 0 to 31

**Example** :DISPLAY:INTENSITY:CURSOR 10  
:DISPLAY:INTENSITY:CURSOR?  
-> :DISPLAY:INTENSITY:CURSOR 10

### **:DISPlay:INTERpolate**

**Function** Sets the display interpolation format or queries the current setting.

**Syntax** :DISPlay:INTERpolate {LINE|OFF|PULSe|SINE}  
:DISPlay:INTERpolate?

**Example** :DISPLAY:INTERPOLATE SINE  
:DISPLAY:INTERPOLATE? -> :DISPLAY:  
INTERPOLATE SINE

### **:DISPlay:MAPPING?**

**Function** Queries all settings related to the waveform mapping to the split screen.

**Syntax** :DISPlay:MAPPING?

**Example** :DISPLAY:MAPPING? -> :DISPLAY:MAPPING:  
MODE AUTO;TRACE1 1;  
TRACE2 1;TRACE3 1;TRACE4 1;TRACE5 1;  
TRACE6 1;TRACE7 1;TRACE8 1

### **:DISPlay:MAPPING[:MODE]**

**Function** Sets the waveform mapping mode for the split screen or queries the current setting.

**Syntax** :DISPlay:MAPPING[:MODE] {AUTO|MANual}  
:DISPlay:MAPPING[:MODE]?

**Example** :DISPLAY:MAPPING:MODE AUTO  
:DISPLAY:MAPPING:MODE? -> :DISPLAY:  
MAPPING:MODE AUTO

### **:DISPlay:MAPPING:TRACe<x>**

**Function** Sets the mapping of the waveform to the split screen or queries the current setting.

**Syntax** :DISPlay:MAPPING:TRACe<x> {<NRf>}  
:DISPlay:MAPPING:TRACe<x>?  
<x> = 1 to 8  
<NRf> = 1 to 4

**Example** :DISPLAY:MAPPING:TRACE1 3  
:DISPLAY:MAPPING:TRACE1? -> :DISPLAY:  
MAPPING:TRACE1 3

### **:DISPlay:TRANslucent**

**Function** Turns ON/OFF the translucent mode or queries the current setting.

**Syntax** :DISPlay:TRANslucent {<Boolean>}  
:DISPlay:TRANslucent?

**Example** :DISPLAY:TRANSLUCENT ON  
:DISPLAY:TRANSLUCENT? -> :DISPLAY:  
TRANSLUCENT 1

## 5.11 FILE Group

### : FILE?

**Function** Queries all settings related to the specified storage medium or internal memory.

**Syntax** :FILE?

**Example** :FILE? -> :FILE:DIRECTORY:  
CDIRECTORY "\Flash Mem\HOME";:FILE:  
INTERNAL:STORE:MEMORY:TRACE 1;:FILE:  
INTERNAL:STORE:DMEMORY:TRACE 1;:FILE:  
INTERNAL:STORE:SETUP:COMMENT1 "WAVE1";  
COMMENT2 "";COMMENT3 "";COMMENT4 "";  
COMMENT5 "";COMMENT6 "";COMMENT7 "";  
COMMENT8 "";COMMENT9 "";COMMENT10 "";  
COMMENT11 "";COMMENT12 "";LOCK1 1;  
LOCK2 0;LOCK3 0;LOCK4 0;LOCK5 0;  
LOCK6 0;LOCK7 0;LOCK8 0;LOCK9 0;  
LOCK10 0;LOCK11 0;LOCK12 0;:FILE:LOAD:  
BINARY:REFERENCE 1;:FILE:LOAD:ZPOLYGON:  
ZONE 1;:FILE:LOAD:ZWAVE:ZONE 1;:FILE:  
SAVE:AHISTOGRAM:ANALYSIS 1;:FILE:SAVE:  
ANAMING DATE;ASCII:  
COMPRESSION DECIMATION;HISTORY ONE;  
LENGTH 2500;TRACE 1;:FILE:SAVE:MEMORY:  
COMPRESSION DECIMATION;HISTORY ONE;  
LENGTH 2500;TRACE 1;:FILE:SAVE:  
COMMENT "THIS IS TEST";DMEMORY:  
TRACE 1;:FILE:SAVE:FFT:ANALYSIS 1;:  
FILE:SAVE:FLOAT:COMPRESSION DECIMATION;  
HISTORY ONE;LENGTH 2500;TRACE1;:FILE:  
SAVE:NAME "DL9000";SBUS:ANALYSIS 1;:  
FILE:SAVE:ZWAVE:ZONE 1

### : FILE:DIRectory?

**Function** Queries all settings related to the specified storage medium.

**Syntax** :FILE:DIRectory?

**Example** :FILE:DIRectory? -> :FILE:DIRECTORY:  
CDIRECTORY "\Flash Mem\HOME"

### : FILE[:DIRectory]:CDIRectory (Change Directory)

**Function** Sets the current directory or queries the current setting.

**Syntax** :FILE[:DIRectory]:CDIRectory {<String>}  
:FILE[:DIRectory]:CDIRectory?  
<String> = Up to 259 characters

**Example** :FILE:DIRECTORY:CDIRECTORY "\Flash Mem\  
DIR1\DIR2"  
(Absolute path designation)  
:FILE:DIRECTORY:CDIRECTORY "DIR2"  
(Relative path designation)  
:FILE:DIRECTORY:CDIRECTORY "\"  
(Root directory designation)  
:FILE:DIRECTORY:CDIRECTORY? -> :FILE:  
DIRECTORY:  
CDIRECTORY "\Flash Mem\DIR1\DIR2"

**Description**

- Data files cannot be saved to the root directory. Specify a save destination drive for the save directory. The following five drives are selectable.
  - Internal hard disk: "HD"
  - Internal memory: "Flash Mem"
  - Network drive: "Network"
  - PC card: "Storage Card<x>"
  - USB storage device: "USB Storage<x>"<x> = 1 to 4 (however, only a total of four "Storage Card" and "USB Storage" designations are supported (including partitions))
- Sets the specified directory the current directory for saving and loading.
- Absolute and relative path designations are possible.
- To specify an absolute path, enter a backslash at the front of the path.
- Relative path to higher level directories is not allowed.

### : FILE[:DIRectory]:FREE?

**Function** Queries the free disk space in bytes on the specified storage medium.

**Syntax** :FILE[:DIRectory]:FREE?

**Example** :FILE:DIRECTORY:FREE? -> :FILE:  
DIRECTORY:FREE 65536

**Description**

- Returns the size of the current directory.
  - If the current directory is the root directory, "0" is returned.

## 5.11 FILE Group

### **:FILE[:DIRectory]:MDIRECTory (Make Directory)**

**Function** Creates a directory under the specified directory. This is an overlap command.

**Syntax** :FILE[:DIRectory]:MDIRECTory {<String>}  
<String> = Up to 259 characters

**Example** :FILE:DIRECTORY:MDIRECTORY "\\Flash Mem\  
DIR1\DIR2"  
(Absolute path designation)  
:FILE:DIRECTORY:MDIRECTORY "DIR2"  
(Relative path designation)

**Description** For details on the <String> parameter, see the description in FILE[:DIRectory]:CDIRECTory.

### **:FILE:INTernal?**

**Function** Queries all settings related to the internal memory.

**Syntax** :FILE:INTernal?

**Example** :FILE:INTERNAL? -> :FILE:INTERNAL:  
STORE:BINARy:TRAcE 1;;  
FILE:INTERNAL:STORE:DMEMOry:TRAcE 1;;  
FILE:INTERNAL:STORE:SEtUP:  
COMMENT1 "WAVE1";COMMENT2 "";  
COMMENT3 "";COMMENT4 "";COMMENT5 "";  
COMMENT6 "";COMMENT7 "";COMMENT8 "";  
COMMENT9 "";COMMENT10 "";COMMENT11 "";  
COMMENT12 "";LOCK1 1;LOCK2 0;LOCK3 0;  
LOCK4 0;LOCK5 0;LOCK6 0;LOCK7 0;  
LOCK8 0;LOCK9 0;LOCK10 0;LOCK11 0;  
LOCK12 0

### **:FILE:INTernal:RECall:DMEMOry[:EXECute]**

**Function** Executes the loading of the accumulated data from the internal memory. This is an overlap command.

**Syntax** :FILE:INTernal:RECall:DMEMOry  
[:EXECute] {<NRf>}  
<NRf> = 1 to 4

**Example** :FILE:INTERNAL:RECALL:DMEMORY:EXECUTE 1

### **:FILE:INTernal:RECall:SEtUp[:EXECute]**

**Function** Executes the loading of the setup data from the internal memory. This is an overlap command.

**Syntax** :FILE:INTernal:RECall:SEtUp  
[:EXECute] {<NRf>}  
<NRf> = 1 to 12

**Example** :FILE:INTERNAL:RECALL:SETUP:EXECUTE 1

### **:FILE:INTernal:STORE?**

**Function** Queries all settings related to the saving to the internal memory.

**Syntax** :FILE:INTernal:STORE?

**Example** :FILE:INTERNAL:STORE? -> :FILE:  
INTERNAL:STORE:BINARy:TRAcE 1;;  
FILE:INTERNAL:STORE:DMEMOry:TRAcE 1;;  
FILE:INTERNAL:STORE:SEtUP:  
COMMENT1 "WAVE1";COMMENT2 "";  
COMMENT3 "";COMMENT4 "";COMMENT5 "";  
COMMENT6 "";COMMENT7 "";COMMENT8 "";  
COMMENT9 "";COMMENT10 "";COMMENT11 "";  
COMMENT12 "";LOCK1 1;LOCK2 0;LOCK3 0;  
LOCK4 0;LOCK5 0;LOCK6 0;LOCK7 0;  
LOCK8 0;LOCK9 0;LOCK10 0;LOCK11 0;  
LOCK12 0

### **:FILE:INTernal:STORE:BINARy?**

**Function** Queries all settings related to the saving of the binary data to the internal memory.

**Syntax** :FILE:INTernal:STORE:BINARy?

**Example** :FILE:INTERNAL:STORE:BINARy?  
-> :FILE:INTERNAL:STORE:BINARy:  
TRAcE 1

### **:FILE:INTernal:STORE:BINARy[:EXECute]**

**Function** Executes the saving of the binary data to the internal memory. This is an overlap command.

**Syntax** :FILE:INTernal:STORE:BINARy  
[:EXECute] {<NRf>}  
<NRf> = 1 to 4

**Example** :FILE:INTERNAL:STORE:BINARy:EXECUTE 1

### **:FILE:INTernal:STORE:BINARy:TRAcE**

**Function** Sets the trace to be saved in binary data to the internal memory or queries the current setting.

**Syntax** :FILE:INTernal:STORE:BINARy:  
TRAcE {<NRf>}  
:FILE:INTernal:STORE:BINARy:TRAcE?  
<NRf> = 1 to 8

**Example** :FILE:INTERNAL:STORE:BINARy:TRAcE 1  
:FILE:INTERNAL:STORE:BINARy:TRAcE?  
-> :FILE:INTERNAL:STORE:BINARy:TRAcE 1

### **:FILE:INTernal:STORE:DMEMOry?**

**Function** Queries all settings related to the saving of the accumulated data to the internal memory.

**Syntax** :FILE:INTernal:STORE:DMEMOry?

**Example** :FILE:INTERNAL:STORE:DMEMORY?  
-> :FILE:INTERNAL:STORE:DMEMORY:TRAcE 1



**: FILE: INTERNAL: STORE: DMEemory [ : EXECute ]**

Function Executes the saving of the accumulated data to the internal memory. This is an overlap command.

Syntax :FILE:INTERNAL:STORE:DMEemory  
[:EXECute] {<NRf>}  
<NRf> = 1 to 4

Example :FILE:INTERNAL:STORE:DMEemory:EXECUTE 1

**: FILE: INTERNAL: STORE: DMEemory: TRAcE**

Function Sets the trace to be saved in accumulated data to the internal memory or queries the current setting.

Syntax :FILE:INTERNAL:STORE:DMEemory:  
TRAcE {<NRf>|ALL|XY1|XY2}  
:FILE:INTERNAL:STORE:DMEemory:TRAcE?  
<NRf> = 1 to 8

Example :FILE:INTERNAL:STORE:DMEemory:TRACE 1  
:FILE:INTERNAL:STORE:DMEemory:TRACE?  
-> :FILE:INTERNAL:STORE:DMEemory:TRACE 1

**: FILE: INTERNAL: STORE: SETUp?**

Function Queries all settings related to the saving of the setup data to the internal memory.

Syntax :FILE:INTERNAL:STORE:SETUp?  
Example :FILE:INTERNAL:STORE:SETUp? ->

```
:FILE:INTERNAL:STORE:SETUp:
COMMENT1 "WAVE1";COMMENT2 "";
COMMENT3 "";COMMENT4 "";COMMENT5 "";
COMMENT6 "";COMMENT7 "";COMMENT8 "";
COMMENT9 "";COMMENT10 "";COMMENT11 "";
COMMENT12 "";LOCK1 1;LOCK2 0;LOCK3 0;
LOCK4 0;LOCK5 0;LOCK6 0;LOCK7 0;
LOCK8 0;LOCK9 0;LOCK10 0;LOCK11 0;
LOCK12 0
```

**: FILE: INTERNAL: STORE: SETUp: COMMENT<x>**

Function Sets the comment to the setup data to be saved to the internal memory or queries the current setting.

Syntax :FILE:INTERNAL:STORE:SETUp:  
COMMENT<x> {<String>}  
:FILE:INTERNAL:STORE:SETUp:COMMENT<x>?  
<x> = 1 to 12  
<String> = Up to 16 characters

Example :FILE:INTERNAL:STORE:SETUp:  
COMMENT1 "WAVE1"  
:FILE:INTERNAL:STORE:SETUp:COMMENT1?  
-> :FILE:INTERNAL:STORE:SETUp:  
COMMENT1 "WAVE1"

**: FILE: INTERNAL: STORE: SETUp: DATE<x>?**

Function Queries the date/time when the setup data was saved to the internal memory.

Syntax :FILE:INTERNAL:STORE:SETUp:DATE<x>?  
<x> = 1 to 12

Example :FILE:INTERNAL:STORE:SETUp:DATE1?  
-> :FILE:INTERNAL:STORE:SETUp:DATE1  
"2007/01/12 14:58:02"

**: FILE: INTERNAL: STORE: SETUp [ : EXECute ]**

Function Executes the saving of the setup data to the internal memory. This is an overlap command.

Syntax :FILE:INTERNAL:STORE:SETUp  
[:EXECute] {<NRf>}  
<NRf> = 1 to 12

Example :FILE:INTERNAL:STORE:SETUp:EXECUTE 1

**: FILE: INTERNAL: STORE: SETUp: LOCK<x>**

Function Turns ON/OFF the read-only attribute of the setup data in the internal memory or queries the current setting.

Syntax :FILE:INTERNAL:STORE:SETUp:  
LOCK<x> {<Boolean>}  
:FILE:INTERNAL:STORE:SETUp:LOCK<x>?  
<x> = 1 to 12

Example :FILE:INTERNAL:STORE:SETUp:LOCK1 ON  
:FILE:INTERNAL:STORE:SETUp:LOCK1?  
-> :FILE:INTERNAL:STORE:SETUp:LOCK1 1

**: FILE: INTERNAL: UNLoad:****DMEemory [ : EXECute ]**

Function Clears the loaded accumulated data.

Syntax :FILE:INTERNAL:UNLoad:DMEemory[:EXECute]  
Example :FILE:INTERNAL:UNLoad:DMEemory:EXECUTE

**: FILE: LOAD?**

Function Queries all settings related to the loading of files on the specified storage medium.

Syntax :FILE:LOAD?  
Example :FILE:LOAD? -> :FILE:LOAD:BINARy:  
REFERENCE 1;:FILE:  
LOAD:ZPOLYgon:ZONE 1;:FILE:LOAD:ZWAVE:  
ZONE 1

**: FILE: LOAD: { BINARy | ZPOLYgon | ZWAVE } ?**

Function Queries all settings related to the loading of specific data.

Syntax :FILE:LOAD:{BINARy|ZPOLYgon|ZWAVE}?  
Example (The following is an example for binary data.)  
:FILE:LOAD:BINARy? -> :FILE:LOAD:  
BINARy:REFERENCE 1

**: FILE: LOAD: { BINARy | DMEemory | MASK | SETUp | ZPOLYgon | ZWAVE } : ABORT**

Function Aborts the loading of specific data.

Syntax :FILE:LOAD:{BINARy|DMEemory|MASK|SETUp|  
ZPOLYgon|ZWAVE}:ABORT

Example (The following is an example for binary data.)  
:FILE:LOAD:BINARy:ABORT

## 5.11 FILE Group

### **:FILE:LOAD:{BINARY|DMEMORY|MASK|SETUP|ZPOLYGON|ZWAVE}[ :EXECUTE]**

**Function** Executes the loading of specific data. This is an overlap command.

**Syntax** :FILE:LOAD:{BINARY|DMEMORY|MASK|SETUP|ZPOLYGON|ZWAVE}[ :EXECUTE] {<String>}  
<String> = Up to 259 characters

**Example** (The following is an example for binary data.)

```
:FILE:LOAD:BINARy:
EXECUTE "\Flash Mem\DIR\DATA"
(Absolute path designation)
:FILE:LOAD:BINARy:EXECUTE "DATA"
(Relative path designation)
```

**Description** For details on the <String> parameter, see the description in FILE[:DIRectory]:CDIRectory.

### **:FILE:LOAD:BINARy:REFEreNce**

**Function** Sets the load destination of the binary data or queries the current setting.

**Syntax** :FILE:LOAD:BINARy:REFEreNce {<NRf>|ACQMemory}  
:FILE:LOAD:BINARy:REFEreNce?  
<NRf> = 1 to 4

**Example** :FILE:LOAD:BINARy:REFEreNce 1  
:FILE:LOAD:BINARy:REFEreNce? -> :FILE:LOAD:BINARy:REFEreNce 1

### **:FILE:LOAD:{ZPOLYGON|ZWAVE}:ZONE**

**Function** Sets the load destination of the zone data of specific data or queries the current setting.

**Syntax** :FILE:LOAD:{ZPOLYGON|ZWAVE}:ZONE {<NRf>}  
:FILE:LOAD:{ZPOLYGON|ZWAVE}:ZONE?  
<NRf> = 1 to 4

**Example** (The following is an example for ZONE WAVE data.)

```
:FILE:LOAD:ZWAVE:ZONE 1
:FILE:LOAD:ZWAVE:ZONE? -> :FILE:LOAD:ZWAVE:ZONE 1
```

### **:FILE:SAVE?**

**Function** Queries all settings related to the saving of data.

**Syntax** :FILE:SAVE?

**Example** :FILE:SAVE? -> :FILE:SAVE:AHISTOGRAM:ANALYSIS 1;:FILE:SAVE:ANAMING DATE;ASCII:COMPRESSION DECIMATION;HISTORY ONE;LENGTH 2500;TRACE 1;:FILE:SAVE:BINARy:COMPRESSION DECIMATION;HISTORY ONE;LENGTH 2500;TRACE 1;:FILE:SAVE:COMMENT "THIS IS TEST";DMEMORY:TRACE 1;:FILE:SAVE:FFT:ANALYSIS 1;:FILE:SAVE:FLOAT:COMPRESSION DECIMATION;HISTORY ONE;LENGTH 2500;TRACE 1;:FILE:SAVE:NAME "DL9000";SBUS:ANALYSIS 1;:FILE:SAVE:ZWAVE:ZONE 1

### **:FILE:SAVE:{AHISTOGRAM|ASCii|BINARy|DMEMORY|FFT|FLOat|SBUS|ZWAVE}?**

**Function** Queries all settings related to the saving of specific data.

**Syntax** :FILE:SAVE:{AHISTOGRAM|ASCii|BINARy|DMEMORY|FFT|FLOat|SBUS|ZWAVE}?

**Example** (The following is an example for binary data.)

```
:FILE:SAVE:BINARy? -> :FILE:SAVE:BINARy:COMPRESSION DECIMATION;HISTORY ONE;LENGTH 2500;TRACE 1
```

### **:FILE:SAVE:{AHISTOGRAM|ASCii|BINARy|DMEMORY|FFT|FLOat|SETUP|SBUS|WPARAMETER|ZWAVE}:ABORT**

**Function** Aborts the save operation of specific data.

**Syntax** :FILE:SAVE:{AHISTOGRAM|ASCii|BINARy|DMEMORY|FFT|FLOat|SETUP|SBUS|WPARAMETER|ZWAVE}:ABORT

**Example** (The following is an example for binary data.)

```
:FILE:SAVE:BINARy:ABORT
```

### **:FILE:SAVE:{AHISTOGRAM|ASCii|BINARy|DMEMORY|FFT|FLOat|SETUP|SBUS|WPARAMETER|ZWAVE}[ :EXECUTE]**

**Function** Executes the saving of specific data by specifying a file name. This is an overlap command.

**Syntax** :FILE:SAVE:{AHISTOGRAM|ASCii|BINARy|DMEMORY|FFT|FLOat|SETUP|SBUS|WPARAMETER|ZWAVE}[ :EXECUTE] {<String>}  
<String> = Up to 259 characters

**Example** (The following is an example for binary data.)

```
:FILE:SAVE:BINARy:EXECUTE
:FILE:SAVE:BINARy:EXECUTE "\Flash Mem\DIR\DATA"
(Absolute path designation)
:FILE:SAVE:BINARy:EXECUTE "DATA"
(Relative path designation)
```

**Description** • If the path is not specified, the data is saved to the file name specified by :FILE:SAVE:NAME.

• If the path is specified, the auto naming function is disabled.

• For details on the <String> parameter, see the description in FILE[:DIRectory]:CDIRectory.

**: FILE: SAVE: {AHIStogram|FFT|SBUS} :****ANALysis**

**Function** Sets the trace of specific data to be saved or queries the current setting.

**Syntax** FILE:SAVE:{AHIStogram|FFT|SBUS}:  
ANALysis {<NRf>}  
FILE:SAVE:{AHIStogram|FFT|SBUS}:  
ANALysis?  
<NRf> = 1 to 2

**Example** (The following is an example for FFT data.)

```
:FILE:SAVE:FFT:ANALYSIS 1
:FILE:SAVE:FFT:ANALYSIS? -> :FILE:SAVE:
FFT:ANALYSIS 1
```

**: FILE: SAVE: ANAMing**

**Function** Sets the type of auto naming of file names of the saved data or queries the current setting.

**Syntax** :FILE:SAVE:ANAMing {DATE|NUMBering|OFF}  
:FILE:SAVE:ANAMing?

**Example** :FILE:SAVE:ANAMING DATE

```
:FILE:SAVE:ANAMING? -> :FILE:SAVE:
ANAMING DATE
```

**: FILE: SAVE: ASCii: RANGE**

**Function** Sets the ASCII data saving range or queries the current setting.

**Syntax** :FILE:SAVE:ASCii:RANGE {MAIN|Z1|Z2}  
:FILE:SAVE:ASCii:RANGE?

**Example** :FILE:SAVE:ASCii:RANGE MAIN

```
:FILE:SAVE:ASCii:RANGE? -> :FILE:SAVE:
ASCii:RANGE MAIN
```

**: FILE: SAVE: {ASCii|BINary|FLOat} :****COMPRESSion**

**Function** Sets the compression method for saving specific data or queries the current setting.

**Syntax** :FILE:SAVE:{ASCii|BINary|FLOat}:  
COMPRESSion {DECimation|OFF|PTOPeak}  
:FILE:SAVE:{ASCii|BINary|FLOat}:  
COMPRESSion?

**Example** (The following is an example for binary data.)

```
:FILE:SAVE:BINARY:
COMPRESSION DECIMATION
:FILE:SAVE:BINARY:COMPRESSION?
-> :FILE:SAVE:BINARY:
COMPRESSION DECIMATION
```

**: FILE: SAVE: {ASCii|BINary|FLOat} :****HISTory**

**Function** Sets whether to save the entire data or the selected data of specific data or queries the current setting.

**Syntax** :FILE:SAVE:{ASCii|BINary|FLOat}:  
HISTory {ALL|ONE}  
:FILE:SAVE:{ASCii|BINary|FLOat}:  
HISTory?

**Example** (The following is an example for binary data.)

```
:FILE:SAVE:BINARY:HISTORY ALL
:FILE:SAVE:BINARY:HISTORY? -> :FILE:
SAVE:BINARY:HISTORY ALL
```

**: FILE: SAVE: {ASCii|BINary|FLOat} : LENGTH**

**Function** Sets the size of waveform data to save for each type of data or queries the current setting.

**Syntax** :FILE:SAVE:{ASCii|BINary|FLOat}:  
LENGTH {<NRf>}  
:FILE:SAVE:{ASCii|BINary|FLOat}:LENGTH?  
<NRf> = See the main unit User's Manual.

**Example** (The following is an example with binary data. )

```
:FILE:SAVE:BINARY:LENGTH 2500
:FILE:SAVE:BINARY:LENGTH? ->
:FILE:SAVE:BINARY:LENGTH 2500
```

**: FILE: SAVE: {ASCii|BINary|FLOat} : TRACE**

**Function** Sets the trace of specific data to be saved or queries the current setting.

**Syntax** :FILE:SAVE:{ASCii|BINary|FLOat}:  
TRACE {<NRf>|ALL}  
:FILE:SAVE:{ASCii|BINary|FLOat}:TRACE?  
<NRf> = 1 to 8

**Example** (The following is an example for binary data.)

```
:FILE:SAVE:BINARY:TRACE 1
:FILE:SAVE:BINARY:TRACE? -> :FILE:SAVE:
BINARY:TRACE 1
```

**: FILE: SAVE: COMMENT**

**Function** Sets the comment of data to be saved or queries the current setting.

**Syntax** :FILE:SAVE:COMMENT {<String>}  
:FILE:SAVE:COMMENT?  
<String> = Up to 160 characters

**Example** :FILE:SAVE:COMMENT "THIS IS TEST"

```
:FILE:SAVE:COMMENT? -> :FILE:SAVE:
COMMENT "THIS IS TEST"
```

## 5.11 FILE Group

---

### **: FILE : SAVE : DMEMoRY : TRACe**

Function Sets the trace to be saved as accumulated data or queries the current setting.

Syntax : FILE:SAVE:DMEMoRY:TRACe {<NRf>|ALL|XY1|XY2}  
: FILE:SAVE:DMEMoRY:TRACe?  
<NRf> = 1 to 8

Example : FILE:SAVE:DMEMoRY:TRACE 1  
: FILE:SAVE:DMEMoRY:TRACE? -> : FILE:  
SAVE:DMEMoRY:TRACE 1

### **: FILE : SAVE : NAME**

Function Sets the name of the data file to be saved or queries the current setting.

Syntax : FILE:SAVE:NAME {<Filename>}  
: FILE:SAVE:NAME?

Example : FILE:SAVE:NAME "DL9000"  
: FILE:SAVE:NAME?  
-> : FILE:SAVE:NAME "DL9000"

### **: FILE : SAVE : ZWAVE : ZONE**

Function Sets the zone of the zone data to be saved or queries the current setting.

Syntax : FILE:SAVE:ZWAVE:ZONE {<NRf>}  
: FILE:SAVE:ZWAVE:ZONE?  
<NRf> = 1 to 4

Example : FILE:SAVE:ZWAVE:ZONE 1  
: FILE:SAVE:ZWAVE:ZONE? -> : FILE:SAVE:  
ZWAVE:ZONE 1

## 5.12 GONogo Group

### **:GONogo?**

**Function** Queries all settings related to GO/NO-GO determination.

**Syntax** :GONogo?

**Example** :GONOGO? -> :GONOGO:ACTION:BUZZER 0;  
HCOPIY 0;MAIL:INTERVAL OFF;MODE 0;;  
GONOGO:ACTION:SAVE 0;;GONOGO:  
CONDITION1 DONTCARE;  
CONDITION2 DONTCARE;  
CONDITION3 DONTCARE;  
CONDITION4 DONTCARE;  
LOGIC AND;MODE OFF;SCONDITION:NGCOUNT  
1;STOPCOUNT 1;;GONOGO:  
TELECOMTEST:SELECT1:MASK:ELEMENT1:  
WCOUNT 1.000E+00,2.000E+00;;GONOGO:  
TELECOMTEST:SELECT2:MASK:ELEMENT1:  
WCOUNT 1.000E+00,2.000E+00;;GONOGO:  
TELECOMTEST:SELECT3:MASK:ELEMENT1:  
WCOUNT 1.000E+00,2.000E+00;;GONOGO:  
TELECOMTEST:SELECT4:MASK:ELEMENT1:  
WCOUNT 2.000E+00,3.000E+00;;GONOGO:  
ZPARAMETER:SELECT1:MODE PARAMETER;  
PARAMETER:CATEGORY FFT;FFT1:  
CALCULATION1 0.000E+00,1.000E+00;;  
GONOGO:ZPARAMETER:SELECT1:PARAMETER:  
MEASURE:STATISTICS MAXIMUM;;GONOGO:  
ZPARAMETER:SELECT1:RECTANGLE:  
HORIZONTAL 0.000E+00,1.000E+00;  
VERTICAL 0.000E+00,1.000E+00;;GONOGO:  
ZPARAMETER:SELECT1:TRACE 1;WAVE:  
TRANGE 1.000E+00,2.000E+00;;GONOGO:  
ZPARAMETER:SELECT1:WINDOW MAIN;;  
GONOGO:ZPARAMETER:SELECT2:  
MODE PARAMETER;PARAMETER:  
CATEGORY FFT;FFT1:  
CALCULATION1 0.000E+00,1.000E+00...

### **:GONogo:ABORT**

**Function** Aborts the GO/NO-GO determination.

**Syntax** :GONogo:ABORT

**Example** :GONOGO:ABORT

### **:GONogo:ACTion?**

**Function** Queries all settings related to the action taken when the determination result is NO-GO and the criteria values.

**Syntax** :GONogo:ACTion?

**Example** :GONOGO:ACTION? -> :GONOGO:ACTION:  
BUZZER 0;HCOPIY 0;MAIL:INTERVAL OFF;  
MODE 0;;GONOGO:ACTION:SAVE 0

### **:GONogo:ACTion:BUZZer**

**Function** Sets whether to sound a buzzer when the determination result is NO-GO or queries the current setting.

**Syntax** :GONogo:ACTion:BUZZer {<Boolean>}  
:GONogo:ACTion:BUZZer?

**Example** :GONOGO:ACTION:BUZZER ON  
:GONOGO:ACTION:BUZZER? -> :GONOGO:  
ACTION:BUZZER 1

### **:GONogo:ACTion:HCOPIy**

**Function** Sets whether to print the screen image on the printer when the determination result is NO-GO or queries the current setting.

**Syntax** :GONogo:ACTion:HCOPIy {<Boolean>}  
:GONogo:ACTion:HCOPIy?

**Example** :GONOGO:ACTION:HCOPIY ON  
:GONOGO:ACTION:HCOPIY? -> :GONOGO:  
ACTION:HCOPIY 1

### **:GONogo:ACTion:MAIL?**

**Function** Queries all settings related to the mail transmission when the determination is NO-GO.

**Syntax** :GONogo:ACTion:MAIL?

**Example** :GONOGO:ACTION:MAIL? -> :GONOGO:  
ACTION:MAIL:INTERVAL 10;MODE 1

### **:GONogo:ACTion:MAIL:INTerval**

**Function** Sets the interval at which to send mail when the determination is NO-GO or queries the current setting.

**Syntax** :GONogo:ACTion:MAIL:INTerval  
{OFF|<Nrf>}  
:GONogo:ACTion:MAIL:INTerval?  
<Nrf> = 1 to 1440 (min)

**Example** :GONOGO:ACTION:MAIL:INTERVAL 10  
:GONOGO:ACTION:MAIL:INTERVAL?  
-> :GONOGO:ACTION:MAIL:INTERVAL 10

### **:GONogo:ACTion:MAIL:MODE**

**Function** Sets whether to send mail when the determination is NO-GO or queries the current setting.

**Syntax** :GONogo:ACTion:MAIL:MODE {<Boolean>}  
:GONogo:ACTion:MAIL:MODE?

**Example** :GONOGO:ACTION:MAIL:MODE ON  
:GONOGO:ACTION:MAIL:MODE? -> :GONOGO:  
ACTION:MAIL:MODE 1

## 5.12 GONogo Group

### **:GONogo:ACTion:SAVE**

**Function** Sets whether to save the waveform data to the storage medium when the determination result is NO-GO or queries the current setting.

**Syntax** :GONogo:ACTion:SAVE {<Boolean>}  
:GONogo:ACTion:SAVE?

**Example** :GONOGO:ACTion:SAVE ON  
:GONOGO:ACTion:SAVE? -> :GONOGO:ACTion:SAVE 1

### **:GONogo:CONDition<x>**

**Function** Sets the GO/NO-GO determination criteria or queries the current setting.

**Syntax** :GONogo:CONDition<x> {DONTcare|IN|OUT}  
:GONogo:CONDition<x>?  
<x> = 1 to 4

**Example** :GONOGO:CONDition1 DONTCARE  
:GONOGO:CONDition1? -> :GONOGO:CONDition1 DONTCARE

### **:GONogo:COUNT?**

**Function** Queries the actual number of GO/NO-GO determinations.

**Syntax** :GONogo:COUNT?

**Example** :GONOGO:COUNT? -> :GONOGO:COUNT 1

### **:GONogo:EXECute**

**Function** Executes the GO/NO-GO determination. This is an overlap command.

**Syntax** :GONogo:EXECute

**Example** :GONOGO:EXECUTE

### **:GONogo:LOGic**

**Function** Sets the GO/NO-GO determination logic or queries the current setting.

**Syntax** :GONogo:LOGic {AND|OR}  
:GONogo:LOGic?

**Example** :GONOGO:LOGic AND  
:GONOGO:LOGic? -> :GONOGO:LOGic AND

### **:GONogo:MODE**

**Function** Sets the GO/NO-GO determination type or queries the current setting.

**Syntax** GONogo:MODE {OFF|TELEcomtest|ZPARAMeter}  
:GONogo:MODE?

**Example** GONOGO:MODE OFF  
:GONOGO:MODE? -> :GONOGO:MODE OFF

### **:GONogo:NGCount?**

**Function** Queries the actual number of NO-GOs of the GO/NO-GO determination.

**Syntax** :GONogo:NGCount?

**Example** :GONOGO:NGCOUNT? -> :GONOGO:NGCOUNT 1

### **:GONogo:SCONdition? (Stop Condition)**

**Function** Queries all settings related to the determination termination condition.

**Syntax** :GONogo:SCONdition?

**Example** :GONOGO:SCONdition? -> :GONOGO:SCONdition:NGCOUNT 1;STOPCOUNT 1

### **:GONogo[:SCONdition]:NGCount**

**Function** Sets the number of NO-GOs that terminates the GO/NO-GO determination or queries the current setting.

**Syntax** :GONogo[:SCONdition]:NGCount {<Nrf>|INFinite}  
:GONogo[:SCONdition]:NGCount?  
<Nrf> = 1 to 1000

**Example** :GONOGO:SCONdition:NGCOUNT 1  
:GONOGO:SCONdition:NGCOUNT? -> :GONOGO:SCONdition:NGCOUNT 1

### **:GONogo[:SCONdition]:STOPcount**

**Function** Sets the acquisition count that terminates the GO/NO-GO determination or queries the current setting.

**Syntax** :GONogo[:SCONdition]:STOPcount {<Nrf>|INFinite}  
:GONogo[:SCONdition]:STOPcount?  
<Nrf> = 1 to 1000000

**Example** :GONOGO:SCONdition:STOPCOUNT 1  
:GONOGO:SCONdition:STOPCOUNT?  
-> :GONOGO:SCONdition:STOPCOUNT 1

### **:GONogo:TELEcomtest?**

**Function** Queries all settings related to telecom test determination.

**Syntax** :GONogo:TELEcomtest?

**Example** :GONOGO:TELECOMTEST? -> :GONOGO:TELECOMTEST:SELECT1:MASK:ELEMENT1:WCount 2.000E+00,1.000E+00;:GONOGO:TELECOMTEST:SELECT2:MASK:ELEMENT1:WCount 2.000E+00,1.000E+00;:GONOGO:TELECOMTEST:SELECT3:MASK:ELEMENT1:WCount 2.000E+00,1.000E+00;:GONOGO:TELECOMTEST:SELECT4:MASK:ELEMENT1:WCount 2.000E+00,1.000E+00

### **:GONogo:TELEcomtest:SELEct<x>?**

**Function** Queries all settings related to the condition of the telecom test determination.

**Syntax** :GONogo:TELEcomtest:SELEct<x>?  
<x> = 1 to 4

**Example** :GONOGO:TELECOMTEST:SELECT1?  
-> :GONOGO:TELECOMTEST:SELECT1:MASK:ELEMENT1:PSPCOUNT 2.000E+00,1.000E+00

**:GONogo:TELEcomtest:SElect<x>:  
EYEPattern?**

Function Queries all settings related to the eye pattern determination of the condition.

Syntax :GONogo:TELEcomtest:SElect<x>:  
EYEPattern?  
<x> = 1 to 4

Example :GONOGO:TELECOMTEST:SELECT1:EYEPATTERN?  
-> :GONOGO:TELECOMTEST:SELECT1:  
EYEPATTERN:DBERATE 2.000E+00,1.000E+00

**:GONogo:TELEcomtest:SElect<x>:  
EYEPattern:<Parameter>**

Function Sets the upper and lower limits of the waveform parameter of the eye pattern determination or queries the current setting.

Syntax :GONogo:TELEcomtest:SElect<x>:  
EYEPattern:<Parameter> {<Nrf>,<Nrf>|  
<Voltage>,<Voltage>|  
<Current>,<Current>|<Time>,<Time>}  
:GONogo:TELEcomtest:SElect<x>:  
EYEPattern:<Parameter>?  
<x> of SElect<x> = 1 to 4  
<Parameter> = {DBERate|EHeight|EWidth|  
FALL|JITter<y>|PCROssing|PDUtYcycle|  
QFActor|RISE|SDBase|SDTop|T1CRossing|  
T2CRossing|VBASe|VCRossing|VTOP}  
<Nrf>, <Voltage>, <Current>, and <Time> = See the  
DL9040/DL9140/DL9240 User's Manual (IM701310-  
01E).  
<y> = 1 or 2

Example The following is an example for  
DBERate.  
:GONOGO:TELECOMTEST:SELECT1:EYEPATTERN:  
DBERATE 1,2  
:GONOGO:TELECOMTEST:SELECT1:EYEPATTERN:  
DBERATE? -> :GONOGO:TELECOMTEST:  
SELECT1:EYEPATTERN:  
DBERATE 2.000E+00,1.000E+00

Description • For the relationship between communication commands and parameters used by the DL9000 Series, see appendix 4.  
• For details on the parameters, see the DL9040/DL9140/DL9240 User's Manual (IM701310-01E).

**:GONogo:TELEcomtest:SElect<x>:MASK?**

Function Queries all settings related to the mask determination of the condition.

Syntax :GONogo:TELEcomtest:SElect<x>:MASK?  
<x> = 1 to 4

Example :GONOGO:TELECOMTEST:SELECT1:MASK?  
-> :GONOGO:TELECOMTEST:SELECT1:MASK:  
ELEMENT1:PSPCOUNT 2.000E+00,1.000E+00

**:GONogo:TELEcomtest:SElect<x>:MASK:  
ELEMENT<x>?**

Function Queries all settings related to the element used in the mask determination.

Syntax :GONogo:TELEcomtest:SElect<x>:MASK:  
ELEMENT<x>?  
<x> of SElect<x> = 1 to 4  
<x> of ELEMENT<x> = 1 to 4

Example :GONOGO:TELECOMTEST:SELECT1:MASK:  
ELEMENT1? -> :GONOGO:TELECOMTEST:  
SELECT1:MASK:ELEMENT1:  
PSPCOUNT 2.000E+00,1.000E+00

**:GONogo:TELEcomtest:SElect<x>:MASK:  
ELEMENT<x>:PSPCount  
(Sample Point Count %)**

Function Sets the upper and lower limits of the error rate for the number of sampled data points of the element or queries the current setting.

Syntax :GONogo:TELEcomtest:SElect<x>:MASK:  
ELEMENT<x>:PSPCount {<Nrf>,<Nrf>}  
:GONogo:TELEcomtest:SElect<x>:MASK:  
ELEMENT<x>:PSPCount?  
<x> of SElect<x> = 1 to 4  
<x> of ELEMENT<x> = 1 to 4  
<Nrf> = 0 to 100 (%)

Example :GONOGO:TELECOMTEST:SELECT1:MASK:  
ELEMENT1:PSPCOUNT 1,2  
:GONOGO:TELECOMTEST:SELECT1:MASK:  
ELEMENT1:PSPCOUNT? ->  
:GONOGO:TELECOMTEST:SELECT1:MASK:  
ELEMENT1:PSPCOUNT 2.000E+00,1.000E+00

**:GONogo:TELEcomtest:SElect<x>:MASK:  
ELEMENT<x>:PWCount (Wave Count %)**

Function Sets the upper and lower limits of the error rate for the number of acquisitions of the element or queries the current setting.

Syntax :GONogo:TELEcomtest:SElect<x>:MASK:  
ELEMENT<x>:PWCount {<Nrf>,<Nrf>}  
:GONogo:TELEcomtest:SElect<x>:MASK:  
ELEMENT<x>:PWCount?  
<x> of SElect<x> = 1 to 4  
<x> of ELEMENT<x> = 1 to 4  
<Nrf> = 0 to 100 (%)

Example :GONOGO:TELECOMTEST:SELECT1:MASK:  
ELEMENT1:PWCOUNT 1,2  
:GONOGO:TELECOMTEST:SELECT1:MASK:  
ELEMENT1:PWCOUNT? -> :GONOGO:  
TELECOMTEST:SELECT1:MASK:  
ELEMENT1:PWCOUNT 2.000E+00,1.000E+00

## 5.12 GONogo Group

### **:GONogo:TELecomtest:SElect<x>:MASK: ELEMent<x>:SPCount (Sample Point Count)**

**Function** Sets the upper and lower limits of the number of sampled data points for the element that results in error or queries the current setting.

**Syntax** :GONogo:TELecomtest:SElect<x>:MASK: ELEMent<x>:SPCount {<NRf>,<NRf>}  
:GONogo:TELecomtest:SElect<x>:MASK: ELEMent<x>:SPCount?  
<x> of SElect<x> = 1 to 4  
<x> of ELEMent<x> = 1 to 4  
<NRf> = See the DL9000 User's Manual.

**Example** :GONOGO:TELECOMTEST:SELECT1:MASK:  
ELEMENT1:SPCOUNT 1,2  
:GONOGO:TELECOMTEST:SELECT1:MASK:  
ELEMENT1:SPCOUNT? -> :GONOGO:  
TELECOMTEST:SELECT1:MASK:  
ELEMENT1:SPCOUNT 2.000E+00,1.000E+00

### **:GONogo:TELecomtest:SElect<x>:MASK: ELEMent<x>:WCOunt (Wave Count)**

**Function** Sets the upper and lower limits of the number of acquisitions for the element that results in error or queries the current setting.

**Syntax** :GONogo:TELecomtest:SElect<x>:MASK: ELEMent<x>:WCOunt {<NRf>,<NRf>}  
:GONogo:TELecomtest:SElect<x>:MASK: ELEMent<x>:WCOunt?  
<x> of SElect<x> = 1 to 4  
<x> of ELEMent<x> = 1 to 4  
<NRf> = See the DL9000 User's Manual.

**Example** :GONOGO:TELECOMTEST:SELECT1:MASK:  
ELEMENT1:WCOUNT 1,2  
:GONOGO:TELECOMTEST:SELECT1:MASK:  
ELEMENT1:WCOUNT? -> :GONOGO:TELECOMTEST:  
SELECT1:MASK:  
ELEMENT1:WCOUNT 2.000E+00,1.000E+00

### **:GONogo:ZPARAMeter?**

**Function** Queries all settings related to zone/parameter determination.

**Syntax** :GONogo:ZPARAMeter?

**Example** :GONOGO:ZPARAMETER? -> :GONOGO:  
ZPARAMETER:SELECT1:  
MODE PARAMETER;PARAMETER:CATEGORY FFT;  
FFT1:CALCULATION1 0.000E+00,1.000E+00;;  
GONOGO:ZPARAMETER:SELECT1:RECTANGLE:  
HORIZONTAL 0.000E+00,1.000E+00;  
VERTICAL 0.000E+00,1.000E+00;;GONOGO:  
ZPARAMETER:SELECT1:TRACE 1;WAVE:  
TRANGE 1.000E+00,2.000E+00;;GONOGO:  
ZPARAMETER:SELECT1:WINDOW MAIN;;GONOGO:  
ZPARAMETER:SELECT2:MODE PARAMETER;  
PARAMETER:CATEGORY FFT;;FFT1:  
CALCULATION1 0.000E+00,1.000E+00;;  
GONOGO:ZPARAMETER:SELECT2:RECTANGLE:  
HORIZONTAL 0.000E+00,1.000E+00;  
VERTICAL 0.000E+00,1.000E+00;;GONOGO:  
ZPARAMETER:SELECT2:TRACE 1;WAVE:  
TRANGE 1.000E+00,2.000E+00;;GONOGO:  
ZPARAMETER:SELECT2:WINDOW MAIN;;GONOGO:  
ZPARAMETER:SELECT3:MODE PARAMETER;  
PARAMETER:CATEGORY FFT;FFT1:  
CALCULATION1 0.000E+00,1.000E+00;.....

### **:GONogo:ZPARAMeter:SElect<x>?**

**Function** Queries all settings related to the condition of the zone/parameter determination.

**Syntax** :GONogo:ZPARAMeter:SElect<x>?  
<x> = 1 to 4

**Example** :GONOGO:ZPARAMETER:SELECT1? -> :GONOGO:  
ZPARAMETER:SELECT1:  
MODE PARAMETER;PARAMETER:CATEGORY FFT;  
FFT1:CALCULATION1 0.000E+00,1.000E+00;;  
GONOGO:ZPARAMETER:SELECT1:RECTANGLE:  
HORIZONTAL 0.000E+00,1.000E+00;  
VERTICAL 0.000E+00,1.000E+00;;GONOGO:  
ZPARAMETER:SELECT1:TRACE 1;WAVE:  
TRANGE 1.000E+00,2.000E+00;;GONOGO:  
ZPARAMETER:SELECT1:WINDOW MAIN

### **:GONogo:ZPARAMeter:SElect<x>:MODE**

**Function** Sets the mode of the condition or queries the current setting.

**Syntax** :GONogo:ZPARAMeter:SElect<x>:MODE {PARa  
meter|POLYgon|RECTangle|WAVE}  
:GONogo:ZPARAMeter:SElect<x>:MODE?  
<x> = 1 to 4

**Example** :GONOGO:ZPARAMETER:SELECT1:MODE WAVE  
:GONOGO:ZPARAMETER:SELECT1:MODE?  
-> :GONOGO:ZPARAMETER:SELECT1:MODE WAVE



**:GONogo:ZPARAMeter:SElect<x>:  
PARAMeter?**

Function Queries all settings related to the condition parameter.

Syntax :GONogo:ZPARAMeter:SElect<x>:PARAMeter?  
<x> = 1 to 4

Example :GONOGO:ZPARAMETER:SELECT1:PARAMETER?  
-> :GONOGO:ZPARAMETER:SELECT1:  
PARAMETER:CATEGORY FFT;PARAMETER:  
CATEGORY FFT;FFT1:  
CALCULATION1 1.000E+00,0.000E+00

**:GONogo:ZPARAMeter:SElect<x>:  
PARAMeter:CATegory**

Function Sets the parameter category or queries the current setting.

Syntax :GONogo:ZPARAMeter:SElect<x>:PARAMeter:  
CATegory {FFT|MEASure|XY}  
:GONogo:ZPARAMeter:SElect<x>:PARAMeter:  
CATegory?  
<x> = 1 to 4

Example :GONOGO:ZPARAMETER:SELECT1:PARAMETER:  
CATEGORY FFT  
:GONOGO:ZPARAMETER:SELECT1:PARAMETER:  
CATEGORY? -> :GONOGO:ZPARAMETER:  
SELECT1:PARAMETER:CATEGORY FFT

Description This command is valid when :MEASURE:MODE  
CYCLE.

**:GONogo:ZPARAMeter:SElect<x>:  
PARAMeter:FFT<x>?**

Function Queries all settings related to the FFT determination.

Syntax :GONogo:ZPARAMeter:SElect<x>:PARAMeter:  
FFT<x>?  
<x> of SElect<x> = 1 to 4  
<x> of FFT<x> = 1 or 2

Example :GONOGO:ZPARAMETER:SELECT1:PARAMETER:  
FFT1? -> :GONOGO:ZPARAMETER:SELECT1:  
PARAMETER:FFT1:PEAK:  
FREQUENCY1 0.000E+00,0.000E+00

**:GONogo:ZPARAMeter:SElect<x>:  
PARAMeter:FFT<x>:CALCulation<x>**

Function Sets the upper and lower limits of the calculation item of the FFT determination or queries the current setting.

Syntax :GONogo:ZPARAMeter:SElect<x>:PARAMeter:  
FFT<x>:CALCulation<x> {<NRf>,<NRf>}  
:GONogo:ZPARAMeter:SElect<x>:PARAMeter:  
FFT<x>:CALCulation<x>?  
<x> of SElect<x> = 1 to 4  
<x> of FFT<x> = 1 or 2  
<x> of CALCulation<x> = 1 to 4  
<NRf> = -4 to 4 (div)

Example :GONOGO:ZPARAMETER:SELECT1:PARAMETER:  
FFT1:CALCULATION1 0,1  
:GONOGO:ZPARAMETER:SELECT1:PARAMETER:  
FFT1:CALCULATION1? -> :GONOGO:  
ZPARAMETER:SELECT1:PARAMETER:  
FFT1:CALCULATION1 1.000E+00,0.000E+00

**:GONogo:ZPARAMeter:SElect<x>:  
PARAMeter:FFT<x>:PEAK?**

Function Queries all settings related to the peak value of the FFT determination.

Syntax :GONogo:ZPARAMeter:SElect<x>:PARAMeter:  
FFT<x>:PEAK?  
<x> of SElect<x> = 1 to 4  
<x> of FFT<x> = 1 or 2

Example :GONOGO:ZPARAMETER:SELECT1:PARAMETER:  
FFT1:PEAK? -> :GONOGO:ZPARAMETER:  
SELECT1:PARAMETER:FFT1:PEAK:  
FREQUENCY1 1.000E+00,0.000E+00

**:GONogo:ZPARAMeter:SElect<x>:  
PARAMeter:FFT<x>:PEAK:DFREquency**

Function Sets the upper and lower limits between the peak frequencies of the FFT determination or queries the current setting.

Syntax :GONogo:ZPARAMeter:SElect<x>:PARAMeter:  
FFT<x>:PEAK:DFREquency {<Frequency>,<Fr  
equency>}  
:GONogo:ZPARAMeter:SElect<x>:PARAMeter:  
FFT<x>:PEAK:DFREquency?  
<x> of SElect<x> = 1 to 4  
<x> of FFT<x> = 1 or 2  
<Frequency> = See the DL9000 User's Manual

Example :GONOGO:ZPARAMETER:SELECT1:PARAMETER:  
FFT1:PEAK:DFREQUENCY 0,1  
:GONOGO:ZPARAMETER:SELECT1:PARAMETER:  
FFT1:PEAK:DFREQUENCY? -> :GONOGO:  
ZPARAMETER:SELECT1:PARAMETER:FFT1:PEAK:  
DFREQUENCY 1.000E+00,0.000E+00

## 5.12 GONogo Group

### **:GONogo:ZPARAMeter:SElect<x>:**

#### **PARAMeter:FFT<x>:PEAK:DV**

**Function** Sets the upper and lower limits between the peak voltages of the FFT determination or queries the current setting.

**Syntax** :GONogo:ZPARAMeter:SElect<x>:PARAMeter:  
FFT<x>:PEAK:DV {<NRf>,<NRf>}  
:GONogo:ZPARAMeter:SElect<x>:PARAMeter:  
FFT<x>:PEAK:DV?  
<x> of SElect<x> = 1 to 4  
<x> of FFT<x> = 1 or 2  
<NRf> = -4 to 4 (div)

**Example** :GONOGO:ZPARAMETER:SELECT1:PARAMETER:  
FFT1:PEAK:DV 0,1  
:GONOGO:ZPARAMETER:SELECT1:PARAMETER:  
FFT1:PEAK:DV? -> :GONOGO:ZPARAMETER:  
SELECT1:PARAMETER:FFT1:PEAK:  
DV 1.000E+00,0.000E+00

### **:GONogo:ZPARAMeter:SElect<x>:**

#### **PARAMeter:FFT<x>:PEAK:FREQuency<x>**

**Function** Sets the upper and lower limits of the peak frequency of the FFT determination or queries the current setting.

**Syntax** :GONogo:ZPARAMeter:SElect<x>:PARAMeter:  
FFT<x>:PEAK:FREQuency<x> {<Frequency>,  
<Frequency>}  
:GONogo:ZPARAMeter:SElect<x>:PARAMeter:  
FFT<x>:PEAK:FREQuency<x>?  
<x> of SElect<x> = 1 to 4  
<x> of FFT<x> = 1 or 2  
<x> of FREQuency<x> = 1 or 2  
<Frequency> = See the DL9000 User's Manual

**Example** :GONOGO:ZPARAMETER:SELECT1:PARAMETER:  
FFT1:PEAK:FREQUENCY1 0,1  
:GONOGO:ZPARAMETER:SELECT1:PARAMETER:  
FFT1:PEAK:FREQUENCY1? -> :GONOGO:  
ZPARAMETER:SELECT1:PARAMETER:FFT1:  
PEAK:FREQUENCY1 1.000E+00,0.000E+00

### **:GONogo:ZPARAMeter:SElect<x>:**

#### **PARAMeter:FFT<x>:PEAK:V<x>**

**Function** Sets the upper and lower limits of the peak voltage of the FFT determination or queries the current setting.

**Syntax** :GONogo:ZPARAMeter:SElect<x>:PARAMeter:  
FFT<x>:PEAK:V<x> {<NRf>,<NRf>}  
:GONogo:ZPARAMeter:SElect<x>:PARAMeter:  
FFT<x>:PEAK:V<x>?  
<x> of SElect<x> = 1 to 4  
<x> of FFT<x> = 1 or 2  
<x> of V<x> = 1 or 2  
<NRf> = -4 to 4 (div)

**Example** :GONOGO:ZPARAMETER:SELECT1:PARAMETER:  
FFT1:PEAK:V1 1,2  
:GONOGO:ZPARAMETER:SELECT1:PARAMETER:  
FFT1:PEAK:V1? -> :GONOGO:ZPARAMETER:  
SELECT1:PARAMETER:FFT1:PEAK:  
V1 2.000E+00,1.000E+00

### **:GONogo:ZPARAMeter:SElect<x>:**

#### **PARAMeter:MEASure?**

**Function** Queries all settings related to the determination using automated measurement of waveform parameters (measure determination).

**Syntax** :GONogo:ZPARAMeter:SElect<x>:PARAMeter:  
MEASure?  
<x> = 1 to 4

**Example** :GONOGO:ZPARAMETER:SELECT1:PARAMETER:  
MEASURE? -> :GONOGO:ZPARAMETER:  
SELECT1:PARAMETER:MEASURE:  
CALCULATION1 2.000E+00,1.000E+00

### **:GONogo:ZPARAMeter:SElect<x>:**

#### **PARAMeter:MEASure:CALCulation<x>**

**Function** Sets the upper and lower limits of the calculation item of the measure determination or queries the current setting.

**Syntax** :GONogo:ZPARAMeter:SElect<x>:PARAMeter:  
MEASure:CALCulation<x> {<NRf>,<NRf>}  
:GONogo:ZPARAMeter:SElect<x>:PARAMeter:  
MEASure:CALCulation<x>?  
<x> of SElect<x> = 1 to 4  
<x> of CALCulation<x> = 1 to 4  
<NRf> = -4 to 4 (div)

**Example** :GONOGO:ZPARAMETER:SELECT1:PARAMETER:  
MEASURE:CALCULATION1 1,2  
:GONOGO:ZPARAMETER:SELECT1:PARAMETER:  
MEASURE:CALCULATION1? -> :GONOGO:  
ZPARAMETER:SELECT1:PARAMETER:MEASURE:  
CALCULATION1 2.000E+00,1.000E+00

**:GONogo:ZPARAMeter:SElect<x>:****PARAMeter:MEASure:STATistics**

**Function** Sets the statistical value of the measure determination or queries the current setting.

**Syntax** :GONogo:ZPARAMeter:SElect<x>:PARAMeter:MEASure:STATistics {MAXimum|MEAN|MINimum|SIGMa}  
:GONogo:ZPARAMeter:SElect<x>:PARAMeter:MEASure:STATistics?  
<x> = 1 to 4

**Example** :GONOGO:ZPARAMETER:SELECT1:PARAMETER:MEASURE:STATISTICS MAXIMUM  
:GONOGO:ZPARAMETER:SELECT1:PARAMETER:MEASURE:STATISTICS? ->  
:GONOGO:ZPARAMETER:SELECT1:PARAMETER:MEASURE:STATISTICS MAXIMUM

**Description** This command is valid when :MEASURE:MODE CYCLE.

**:GONogo:ZPARAMeter:SElect<x>:****PARAMeter:MEASure:TRACe<x>?**

**Function** Queries all settings related to the trace of the measure determination.

**Syntax** :GONogo:ZPARAMeter:SElect<x>:PARAMeter:MEASure:TRACe<x>?  
<x> of SElect<x> = 1 to 4  
<x> of TRACe<x> = 1 to 8

**Example** :GONOGO:ZPARAMETER:SELECT1:PARAMETER:MEASURE:TRACE1? -> :GONOGO:ZPARAMETER:SELECT1:PARAMETER:MEASURE:TRACE1:AREA1:TYPE:MAXIMUM 2.000E+00,1.000E+00

**:GONogo:ZPARAMeter:SElect<x>:****PARAMeter:MEASure:TRACe<x>:AREA<x>?**

**Function** Queries all settings related to the area of the measure determination.

**Syntax** :GONogo:ZPARAMeter:SElect<x>:PARAMeter:MEASure:TRACe<x>:AREA<x>?  
<x> of SElect<x> = 1 to 4  
<x> of TRACe<x> = 1 to 8  
<x> of AREA<x> = 1 or 2

**Example** :GONOGO:ZPARAMETER:SELECT1:PARAMETER:MEASURE:TRACE1:AREA1? -> :GONOGO:ZPARAMETER:SELECT1:PARAMETER:MEASURE:TRACE1:AREA1:TYPE:MAXIMUM 2.000E+00,1.000E+00

**:GONogo:ZPARAMeter:SElect<x>:****PARAMeter:MEASure:TRACe<x>:AREA<x>:TYPE?**

**Function** Queries all settings related to the waveform parameters of the measure determination.

**Syntax** :GONogo:ZPARAMeter:SElect<x>:PARAMeter:MEASure:TRACe<x>:AREA<x>:TYPE?  
<x> of SElect<x> = 1 to 4  
<x> of TRACe<x> = 1 to 8  
<x> of AREA<x> = 1 or 2

**Example** :GONOGO:ZPARAMETER:SELECT1:PARAMETER:MEASURE:TRACE1:AREA1:TYPE? -> :GONOGO:ZPARAMETER:SELECT1:PARAMETER:MEASURE:TRACE1:AREA1:TYPE:MAXIMUM 2.000E+00,1.000E+00

**:GONogo:ZPARAMeter:SElect<x>:****PARAMeter:MEASure:TRACe<x>:AREA<x>:TYPE:<Parameter>**

**Function** Sets the upper and lower limits of the waveform parameter of the measure determination or queries the current setting.

**Syntax** :GONogo:ZPARAMeter:SElect<x>:PARAMeter:MEASure:TRACe<x>:AREA<x>:TYPE:<Parameter> {<NRf>,<NRf>|<Voltage>,<Voltage>|<Current>,<Current>|<Time>,<Time>|<Frequency>,<Frequency>}  
:GONogo:ZPARAMeter:SElect<x>:PARAMeter:MEASure:TRACe<x>:AREA<x>:TYPE:<Parameter>?  
<x> of SElect<x> = 1 to 4  
<x> of TRACe<x> = 1 to 8  
<x> of AREA<x> = 1 or 2  
<Parameter> = {BURSt|CMEan|COUNT|CRMS|CSDeviation|DElay|DT|DUTYcycle|FALL|FREquency|HIGH|HILow|LOW|MAXimum|MEAN|MINimum|NOVershoot|NWIDTH|PERFrequency|PERiod|POVershoot|PTOPeak|PWIDTH|RISE|RMS|SDEViation|TYCInteg|TYINtegr|V1|V2}<NRf>,<Voltage>,<Current>,<Time>,<Frequency> = See the DL9000 User's Manual.

**Example** (The following is an example for the maximum value of trace 1 and area 1.)  
:GONOGO:ZPARAMETER:SELECT1:PARAMETER:MEASURE:TRACE1:AREA1:TYPE:MAXIMUM 1,2  
:GONOGO:ZPARAMETER:SELECT1:PARAMETER:MEASURE:TRACE1:AREA1:TYPE:MAXIMUM?  
-> :GONOGO:ZPARAMETER:SELECT1:PARAMETER:MEASURE:TRACE1:AREA1:TYPE:MAXIMUM 2.000E+00,1.000E+00

**Description** This command applies to cycle statistical processing when :MEASURE:MODE CYCLE.

## 5.12 GONogo Group

### **:GONogo:ZPARAMeter:SElect<x>:**

#### **PARAMeter:XY<x>?**

Function Queries all settings related to the XY determination.

Syntax :GONogo:ZPARAMeter:SElect<x>:PARAMeter:  
XY<x>?

<x> of SElect<x> = 1 to 4

<x> of XY<x> = 1 or 2

Example :GONOGO:ZPARAMETER:SELECT1:PARAMETER:  
XY1? -> :GONOGO:ZPARAMETER:SELECT1:  
PARAMETER:XY1:  
XYINTEG 2.000E+00,1.000E+00

### **:GONogo:ZPARAMeter:SElect<x>:**

#### **PARAMeter:XY<x>:XYINteG**

Function Sets the upper and lower limits integral value of the XY determination or queries the current setting.

Syntax :GONogo:ZPARAMeter:SElect<x>:PARAMeter:  
XY<x>:XYINteG {<NRf>,<NRf>}

:GONogo:ZPARAMeter:SElect<x>:PARAMeter:  
XY<x>:XYINteG?

<x> of SElect<x> = 1 to 4

<x> of XY<x> = 1 or 2

<NRf> = -4 to 4 (div)

Example :GONOGO:ZPARAMETER:SELECT1:PARAMETER:  
XY1:XYINTEG 1,2  
:GONOGO:ZPARAMETER:SELECT1:PARAMETER:  
XY1:XYINTEG? -> :GONOGO:ZPARAMETER:  
SELECT1:PARAMETER:XY1:  
XYINTEG 2.000E+00,1.000E+00

### **:GONogo:ZPARAMeter:SElect<x>:**

#### **RECTangle?**

Function Queries all settings related to the rectangle determination.

Syntax :GONogo:ZPARAMeter:SElect<x>:RECTangle?  
<x> = 1 to 4

Example :GONOGO:ZPARAMETER:SELECT1:RECTANGLE?  
-> :GONOGO:ZPARAMETER:SELECT1:  
RECTANGLE:HORIZONTAL 1.000E+00,  
0.000E+00;VERTICAL 1.000E+00,0.000E+00

### **:GONogo:ZPARAMeter:SElect<x>:**

#### **RECTangle:HORizontal**

Function Sets the horizontal position of the rectangle used in the rectangle determination or queries the current setting.

Syntax :GONogo:ZPARAMeter:SElect<x>:RECTangle:  
HORizontal {<NRf>,<NRf>}

:GONogo:ZPARAMeter:SElect<x>:RECTangle:  
HORizontal?

<x> = 1 to 4

<NRf> = -5 to 5 div

Example :GONOGO:ZPARAMETER:SELECT1:RECTANGLE:  
HORIZONTAL 0,1  
:GONOGO:ZPARAMETER:SELECT1:RECTANGLE:  
HORIZONTAL? -> :GONOGO:ZPARAMETER:  
SELECT1:RECTANGLE:  
HORIZONTAL 1.000E+00,0.000E+00

### **:GONogo:ZPARAMeter:SElect<x>:**

#### **RECTangle:VERTical**

Function Sets the vertical position of the rectangle used in the rectangle determination or queries the current setting.

Syntax :GONogo:ZPARAMeter:SElect<x>:RECTangle:  
VERTical {<NRf>,<NRf>}

:GONogo:ZPARAMeter:SElect<x>:RECTangle:  
VERTical?

<x> = 1 to 4

<NRf> = -4 to 4 (div)

Example :GONOGO:ZPARAMETER:SELECT1:RECTANGLE:  
VERTICAL 0,1  
:GONOGO:ZPARAMETER:SELECT1:RECTANGLE:  
VERTICAL? -> :GONOGO:ZPARAMETER:  
SELECT1:RECTANGLE:  
VERTICAL 1.000E+00,0.000E+00

### **:GONogo:ZPARAMeter:SElect<x>:TRACe**

Function Sets the source trace of the zone/parameter determination or queries the current setting.

Syntax :GONogo:ZPARAMeter:SElect<x>:TRACe  
{<NRf>}

:GONogo:ZPARAMeter:SElect<x>:TRACe?  
<x> = 1 to 4

<NRf> = 1 to 8

Example :GONOGO:ZPARAMETER:SELECT1:TRACE 1  
:GONOGO:ZPARAMETER:SELECT1:TRACE?  
-> :GONOGO:ZPARAMETER:SELECT1:TRACE 1

### **:GONogo:ZPARAMeter:SElect<x>:WAVE?**

Function Queries all settings related to the wave determination.

Syntax :GONogo:ZPARAMeter:SElect<x>:WAVE?  
<x> = 1 to 4

Example :GONOGO:ZPARAMETER:SELECT1:WAVE?  
-> :GONOGO:ZPARAMETER:SELECT1:WAVE:  
TRANGE 2.000E+00,1.000E+00

**: GONogo: ZPARAMeter: SElect<x>: WAVE:****EDIT<x>: EXIT**

Function Exits the edit menu of the wave determination zone.

Syntax :GONogo:ZPARAMeter:SElect<x>:WAVE:

EDIT<x>:EXIT {<NRf>|QUIT}

<x> of SElect<x> = 1 to 4

<x> of EDIT<x> = 1 to 13 (1 to 8 are traces and 9 to 13 are internal memories)

<NRf> = 1 to 4 (internal memories)

Example :GONOGO:ZPARAMETER:SELECT1:WAVE:EDIT1:  
EXIT 1

Description • An error occurs if this command is issued when the zone is not being edited.

- Saves the zone waveform to the internal memory specified by <NRf>.

**: GONogo: ZPARAMeter: SElect<x>: WAVE:****EDIT<x>: PART**

Function Editing the portion of the zone of the wave determination.

Syntax :GONogo:ZPARAMeter:SElect<x>:WAVE:

EDIT<x>:PART {<NRf>, <NRf>, <NRf>, <NRf>}

<x> of SElect<x> = 1 to 4

<x> of EDIT<x> = 1 to 13 (1 to 8 are traces. 9 to 12 are internal memories. 13 is the zone waveform that is currently displayed)

<NRf> = -5 to 5 (div: partial editing time axis cursor 1 and 2)

-8 to 8 (div: up and down) (in the order partial editing time axis cursor 1, 2, ↑, and ↓)

Example :GONOGO:ZPARAMETER:SELECT1:WAVE:EDIT1:  
PART 1,2,3,4

Description The amount of movement upward or downward is a relative value with respect to the zone waveform that is currently displayed.

**: GONogo: ZPARAMeter: SElect<x>: WAVE:****EDIT<x>: WHOLE**

Function Sets the editing of the entire zone of the waveform zone.

Syntax :GONogo:ZPARAMeter:SElect<x>:WAVE:

EDIT<x>:WHOLE {<NRf>, <NRf>, <NRf>, <NRf>}

<x> of SElect<x> = 1 to 4

<x> of EDIT<x> = 1 to 13 (1 to 8 are traces. 9 to 12 are internal memories. 13 is the zone waveform that is currently displayed)

<NRf> = 0 to 5 (div: left and right)

0 to 8 (div: up and down)

(In the order ←, →, ↑, and ↓)

Example :GONOGO:ZPARAMETER:SELECT1:WAVE:EDIT1:  
WHOLE 1,2,3,4

Description The amount of movement is a relative value with respect to the reference waveform.

**: GONogo: ZPARAMeter: SElect<x>: WAVE:****TRANge**

Function Sets the determination range of the zone determination or queries the current setting.

Syntax :GONogo:ZPARAMeter:SElect<x>:WAVE:

TRANge {<NRf>, <NRf>}

:GONogo:ZPARAMeter:SElect<x>:WAVE:

TRANge?

<x> = 1 to 4

<NRf> = -5 to 5 (div)

Example :GONOGO:ZPARAMETER:SELECT1:WAVE:

TRANGE 1,2

:GONOGO:ZPARAMETER:SELECT1:WAVE:TRANGE?

-> :GONOGO:ZPARAMETER:SELECT1:WAVE:

TRANGE 2.000E+00,1.000E+00

**: GONogo: ZPARAMeter: SElect<x>: WINDow**

Function Sets the source window of the zone determination or queries the current setting.

Syntax :GONogo:ZPARAMeter:SElect<x>:WINDow

{MAIN|XY1|XY2|Z1|Z2}

:GONogo:ZPARAMeter:SElect<x>:WINDow?

<x> = 1 to 4

Example :GONOGO:ZPARAMETER:SELECT1:WINDOW MAIN

:GONOGO:ZPARAMETER:SELECT1:WINDOW?

-> :GONOGO:ZPARAMETER:SELECT1:

WINDOW MAIN

## 5.13 HCOpy Group

### **:HCOpy? (Hard COPY)**

**Function** Queries all settings related to the output of screen data.

**Syntax** :HCOpy?

**Example** :HCOpy? -> :HCOpy:DIRECTION EXTPRINTER;  
EXTPRINTER:TONE 1;TYPE EINKJET;;HCOpy:  
FILE:FORMAT BMP;SAVE:ANAMING DATE;  
CDIRECTORY "\Flash Mem\DIR1\DIR2";  
NAME "SAMPLE";:HCOpy:FILE:TONE COLOR;;  
HCOpy:NETPRINT:TONE 1;TYPE HINKJET;;  
HCOpy:PRINTER:HRMODE 1

### **:HCOpy:ABORT**

**Function** Aborts data output and paper feeding.

**Syntax** :HCOpy:ABORT

**Example** :HCOpy:ABORT

**Description** This command is valid for :HCOpy:  
DIREction EXTPrinter|PRINter.

### **:HCOpy:DIREction**

**Function** Sets the data output destination or queries the current setting.

**Syntax** :HCOpy:DIREction {EXTPrinter|FILE|  
NETPrint|PRINter}  
:HCOpy:DIREction?

**Example** :HCOpy:DIRECTION EXTPRINTER  
:HCOpy:DIRECTION? -> :HCOpy:DIRECTION  
EXTPRINTER

### **:HCOpy:EXECute**

**Function** Executes the data output. This is an overlap command.

**Syntax** :HCOpy:EXECute {<String>}  
<String> = Up to 259 characters

**Example** :HCOpy:EXECUTE  
:HCOpy:EXECUTE "\Flash Mem\DIR\DATA"  
(Absolute path designation)  
:HCOpy:EXECUTE "DATA"  
(Relative path designation)

**Description**

- If the path is not specified, the data is saved to the file name specified by :HCOpy:FILE:SAVE:NAME.
- If the path is specified, the auto naming function is disabled.
- For details on the <String> parameter, see the description in :HCOpy:FILE:SAVE:CDIRECTory.

### **:HCOpy:EXTPrinter?**

**Function** Queries all settings related to the external printer output.

**Syntax** :HCOpy:EXTPrinter?

**Example** :HCOpy:EXTPRINTER? -> :HCOpy:EXTPRINTER:  
TONE 1;TYPE EINKJET

### **:HCOpy:EXTPrinter:TONE**

**Function** Sets the half tone of the external printer output or queries the current setting.

**Syntax** :HCOpy:EXTPrinter:TONE {<Boolean>}  
:HCOpy:EXTPrinter:TONE?

**Example** :HCOpy:EXTPRINTER:TONE ON  
:HCOpy:EXTPRINTER:TONE? -> :HCOpy:  
EXTPRINTER:TONE 1

### **:HCOpy:EXTPrinter:TYPE**

**Function** Sets the type of output commands to send to the external printer or queries the current setting.

**Syntax** :HCOpy:EXTPrinter:TYPE {EINKjet|  
HINKjet}  
:HCOpy:EXTPrinter:TYPE?

**Example** :HCOpy:EXTPRINTER:TYPE EINKJET  
:HCOpy:EXTPRINTER:TYPE? -> :HCOpy:  
EXTPRINTER:TYPE EINKJET

### **:HCOpy:FILE?**

**Function** Queries all settings related to file output.

**Syntax** :HCOpy:FILE?

**Example** :HCOpy:FILE? -> :HCOpy:FILE:FORMAT BMP;  
SAVE:ANAMING DATE;  
CDIRECTORY "\Flash Mem\DIR1\DIR2";  
NAME "SAMPLE";:HCOpy:FILE:TONE COLOR

### **:HCOpy:FILE:FORMat**

**Function** Sets the file output image format or queries the current setting.

**Syntax** :HCOpy:FILE:FORMat {BMP|JPEG|PNG}  
:HCOpy:FILE:FORMat?

**Example** :HCOpy:FILE:FORMAT BMP  
:HCOpy:FILE:FORMAT? -> :HCOpy:FILE:  
FORMAT BMP

### **:HCOpy:FILE:SAVE?**

**Function** Queries all settings related to the saving of file output.

**Syntax** :HCOpy:FILE:SAVE?

**Example** :HCOpy:FILE:SAVE? -> :HCOpy:FILE:SAVE:  
ANAMING DATE;  
CDIRECTORY "\Flash Mem\DIR1\DIR2";  
NAME "SAMPLE"

**: HCOpy: FILE: SAVE: ANAMing**

**Function** Sets the type of auto naming of save destination file names or queries the current setting.

**Syntax** :HCOpy:FILE:SAVE:ANAMing {DATE|NUMBERing|OFF}  
:HCOpy:FILE:SAVE:ANAMing?

**Example** :HCOpy:FILE:SAVE:ANAMING DATE  
:HCOpy:FILE:SAVE:ANAMING? -> :HCOpy:FILE:SAVE:ANAMING DATE

**: HCOpy: FILE: SAVE: CDIRECTory (Change Directory)**

**Function** Sets the save destination directory name or queries the current setting.

**Syntax** :HCOpy:FILE:SAVE:CDIRECTory {<String>}  
:HCOpy:FILE:SAVE:CDIRECTory?  
<String> = Up to 259 characters

**Example** :HCOpy:FILE:SAVE:CDIRECTORY "\Flash Mem\DIR1\DIR2"  
(Absolute path designation)  
:HCOpy:FILE:SAVE:CDIRECTORY "DIR2"  
(Relative path designation)  
:HCOpy:FILE:SAVE:CDIRECTORY "\"  
(Root directory designation)  
:HCOpy:FILE:SAVE:CDIRECTORY? -> :HCOpy:FILE:SAVE:CDIRECTORY "\Flash Mem\DIR1\DIR2"

**Description** • Data files cannot be saved to the root directory. Specify a save destination drive for the current directory. The following five drives are selectable.

- Internal hard disk: "HD"
- Internal memory: "Flash Mem"
- Network drive: "Network"
- PC card: "Storage Card<x>"
- USB storage device: "USB Storage<x>"

<x> = 1 to 4 (however, only a total of four "Storage Card" and "USB Storage" designations are supported (including partitions))

- Sets the specified directory the current directory for saving and loading.
- Absolute and relative path designations are possible.
- To specify an absolute path, enter a backslash at the front of the path.
- Relative path to higher level directories is not allowed.

**: HCOpy: FILE: SAVE: NAME**

**Function** Sets the save destination file name or queries the current setting.

**Syntax** :HCOpy:FILE:SAVE:NAME <Filename>  
:HCOpy:FILE:SAVE:NAME?

**Example** :HCOpy:FILE:SAVE:NAME "SAMPLE"  
:HCOpy:FILE:SAVE:NAME? -> :HCOpy:FILE:SAVE:NAME "SAMPLE"

**: HCOpy: NETPrint?**

**Function** Queries all settings related to network printer output or queries the current setting.

**Syntax** :HCOpy:NETPrint?  
**Example** :HCOpy:NETPRINT? -> :HCOpy:NETPRINT: TONE 1;TYPE HINKJET

**: HCOpy: NETPrint: TONE**

**Function** Sets the half tone for the network printer or queries the current setting.

**Syntax** :HCOpy:NETPrint:TONE {<Boolean>}  
:HCOpy:NETPrint:TONE?

**Example** :HCOpy:NETPRINT:TONE ON  
:HCOpy:NETPRINT:TONE? -> :HCOpy:NETPRINT:TONE 1

**Description** Cannot be turned {ON} when :HCOpy:NETPrint:TYPE HLASer is used.

**: HCOpy: NETPrint: TYPE**

**Function** Sets the output command type for the network printer or queries the current setting.

**Syntax** :HCOpy:NETPrint:TYPE {HINKjet|HLASer}  
:HCOpy:NETPrint:TYPE?

**Example** :HCOpy:NETPRINT:TYPE HINKJET  
:HCOpy:NETPRINT:TYPE? -> :HCOpy:NETPRINT:TYPE HINKJET

**: HCOpy: PRINter?**

**Function** Queries all settings related to the built-in printer output.

**Syntax** :HCOpy:PRINter?  
**Example** :HCOpy:PRINTER? -> :HCOpy:PRINTER: HRMODE 1

**: HCOpy: PRINter: HRMode**

**Function** Turns ON/OFF the high resolution mode of the built-in printer output or queries the current setting.

**Syntax** :HCOpy:PRINter:HRMode {<Boolean>}  
:HCOpy:PRINter:HRMode?

**Example** :HCOpy:PRINTER:HRMODE ON  
:HCOpy:PRINTER:HRMODE? -> :HCOpy:PRINTER:HRMODE 1

## 5.14 HISTory Group

### **:HISTory?**

**Function** Queries all settings related to the history function.

**Syntax** :HISTory?

**Example** :HISTORY? -> :HISTORY:CURRENT:  
DISPLAY 0,-10;DMODE ONE;MODE RECORD;  
RECORD 0;REPLAY:SPEED 1;:HISTORY:  
CURRENT:SEARCH:LOGIC AND;SELECT1:  
CONDITION IN;MODE PARAMETER;PARAMETER:  
CATEGORY MEASURE;MEASURE:  
CALCULATION1 1.000E+00,2.000E+00;;  
HISTORY:CURRENT:SEARCH:SELECT1:  
RECTANGLE:HORIZONTAL 1.000E+00,  
2.000E+00;VERTICAL 1.000E+00,  
2.000E+00;;:HISTORY:CURRENT:SEARCH:  
SELECT1:TRACE 1;:HISTORY:CURRENT:  
SEARCH:SELECT1:WAVE:TRANGE 1.000E+00,  
2.000E+00;;:HISTORY:CURRENT:SEARCH:  
SELECT1:WINDOW MAIN;:HISTORY:CURRENT:  
SEARCH:SELECT2:CONDITION IN;  
MODE PARAMETER;PARAMETER:  
CATEGORY MEASURE;MEASURE:  
CALCULATION1 1.000E+00,2.000E+00;;  
HISTORY:CURRENT:SEARCH:SELECT2:  
RECTANGLE:HORIZONTAL 1.000E+00,  
2.000E+00;VERTICAL 1.000E+00,  
2.000E+00;;:HISTORY:CURRENT:SEARCH:  
SELECT2:TRACE 1;:HISTORY:CURRENT:  
SEARCH:SELECT2:WAVE:TRANGE 1.000E+00,  
2.000E+00;;:HISTORY:CURRENT:SEARCH:  
SELECT2:WINDOW MAIN;:HISTORY:CURRENT:  
SEARCH;SELECT3:CONDITION IN;  
MODE PARAMETER;PARAMETER:  
CATEGORY MEASURE;MEASURE:  
CALCULATION1 1.000E+00,2.000E+00.....

### **:HISTory:CURRent?**

**Function** Queries all settings related to the history function of the current waveform (CH1 to 4, M1 to 8).

**Syntax** :HISTory:CURRent?

**Example** :HISTORY:CURRENT? -> :HISTORY:CURRENT:  
DISPLAY 0,-10;DMODE ONE;MODE RECORD;  
RECORD 0;REPLAY:SPEED 1;:HISTORY:  
CURRENT:SEARCH:LOGIC AND;SELECT1:  
CONDITION IN;MODE PARAMETER;PARAMETER:  
CATEGORY MEASURE;MEASURE:  
CALCULATION1 1.000E+00,2.000E+00;;  
HISTORY:CURRENT:SEARCH:SELECT1:  
RECTANGLE:HORIZONTAL 1.000E+00,  
2.000E+00;VERTICAL 1.000E+00,  
2.000E+00;;:HISTORY:CURRENT:SEARCH:  
SELECT1:TRACE 1;:HISTORY:CURRENT:  
SEARCH:SELECT1:WAVE:TRANGE 1.000E+00,  
2.000E+00;;:HISTORY:CURRENT:SEARCH:  
SELECT1:WINDOW MAIN;:HISTORY:CURRENT:  
SEARCH:SELECT2:CONDITION IN;  
MODE PARAMETER;PARAMETER:  
CATEGORY MEASURE;MEASURE:  
CALCULATION1 1.000E+00,2.000E+00;;  
HISTORY:CURRENT:SEARCH:SELECT2:  
RECTANGLE:HORIZONTAL 1.000E+00,  
2.000E+00;VERTICAL 1.000E+00,  
2.000E+00;;:HISTORY:CURRENT:SEARCH:  
SELECT2:TRACE 1;:HISTORY:CURRENT:  
SEARCH:SELECT2:WAVE:TRANGE 1.000E+00,  
2.000E+00;;:HISTORY:CURRENT:SEARCH:  
SELECT2:WINDOW MAIN;:HISTORY:CURRENT:  
SEARCH;SELECT3:CONDITION IN;  
MODE PARAMETER;PARAMETER:  
CATEGORY MEASURE;MEASURE:  
CALCULATION1 1.000E+00,2.000E+00.....

### **:HISTory[:CURRent]:DISPlay**

**Function** Sets the start number and end number of the display record of the history waveform or queries the current setting.

**Syntax** :HISTory[:CURRent]:DISPlay  
{<NRf>,<NRf>}  
:HISTory[:CURRent]:DISPlay?  
<NRf> = See the DL9000 User's Manual.

**Example** :HISTORY:CURRENT:DISPLAY 0,-10  
:HISTORY:CURRENT:DISPLAY? -> :HISTORY:  
CURRENT:DISPLAY 0,-10



**:HISToRY[:CURRENT]:DMODE (Display Mode)**

**Function** Sets the display mode of the history waveform or queries the current setting.

**Syntax** :HISToRY[:CURRENT]:DMODE {ACOLor|AHTone|AINTensity|ONE}  
:HISToRY[:CURRENT]:DMODE?

**Example** :HISTORY:CURRENT:DMODE ONE  
:HISTORY:CURRENT:DMODE? -> :HISTORY:CURRENT:DMODE ONE

**:HISToRY[:CURRENT]:MODE**

**Function** Sets the highlight display mode of the history waveform or queries the current setting.

**Syntax** :HISToRY[:CURRENT]:MODE {AVERAge|RECOrd}  
:HISToRY[:CURRENT]:MODE?

**Example** :HISTORY:CURRENT:MODE RECORD  
:HISTORY:CURRENT:MODE? -> :HISTORY:CURRENT:MODE RECORD

**:HISToRY[:CURRENT]:RECOrd**

**Function** Sets the target record of the history waveform or queries the current setting.

**Syntax** :HISToRY[:CURRENT]:RECOrd {<NRf>|MINimum}  
:HISToRY[:CURRENT]:RECOrd?  
<NRf> = See the DL9000 User's Manual.

**Example** :HISTORY:CURRENT:RECORD 0  
:HISTORY:CURRENT:RECORD? -> :HISTORY:CURRENT:RECORD 0

**Description** Specifying MINimum sets the record to the minimum record number.

**:HISToRY[:CURRENT]:RECOrd? MINimum**

**Function** Queries the minimum record number of the history waveform.

**Syntax** :HISToRY[:CURRENT]:RECOrd? MINimum

**Example** :HISTORY:CURRENT:RECORD? MINIMUM  
-> :HISTORY:CURRENT:RECORD -1

**Description** Specifying MINimum sets the record to the minimum record number.

**:HISToRY[:CURRENT]:REPLay?**

**Function** Queries all settings related to the replay function of the history function.

**Syntax** :HISToRY[:CURRENT]:REPLay?

**Example** :HISTORY:CURRENT:REPLAY? -> :HISTORY:CURRENT:REPLAY:SPEED 1

**:HISToRY[:CURRENT]:REPLay:JUMP**

**Function** Jumps the history waveform to the specified record number.

**Syntax** :HISToRY[:CURRENT]:REPLay:JUMP {MAXimum|MINimum}

**Example** :HISTORY:CURRENT:REPLAY:JUMP MAXIMUM

**:HISToRY[:CURRENT]:REPLay:SPEEd**

**Function** Sets the replay speed of the history waveform or queries the current setting.

**Syntax** :HISToRY[:CURRENT]:REPLay:SPEEd {<NRf>|PER3|PER10|PER30|PER60}  
:HISToRY[:CURRENT]:REPLay:SPEEd?  
<NRf> = 1, 3, 10

**Example** :HISTORY:CURRENT:REPLAY:SPEED 1  
:HISTORY:CURRENT:REPLAY:SPEED?  
-> :HISTORY:CURRENT:REPLAY:SPEED 1

**:HISToRY[:CURRENT]:REPLay:START**

**Function** Starts the replay of the history waveform in the specified direction.

**Syntax** :HISToRY[:CURRENT]:REPLay:START {MAXimum|MINimum}

**Example** :HISTORY:CURRENT:REPLAY:START MAXIMUM

**:HISToRY[:CURRENT]:REPLay:STOP**

**Function** Stops the replay of the history waveform.

**Syntax** :HISToRY[:CURRENT]:REPLay:STOP

**Example** :HISTORY:CURRENT:REPLAY:STOP

## 5.14 HISTory Group

### **:HISTory[:CURRENT]:SEARCH?**

**Function** Queries all settings related to the history search function.

**Syntax** :HISTory[:CURRENT]:SEARCH?

**Example** :HISTORY:CURRENT:SEARCH? -> :HISTORY:CURRENT:SEARCH:LOGIC AND;  
 SELECT1:CONDITION IN;MODE PARAMETER;  
 PARAMETER:CATEGORY MEASURE;MEASURE:  
 CALCULATION1 1.000E+00,2.000E+00;;  
 HISTORY:CURRENT:SEARCH:SELECT1:  
 RECTANGLE:HORIZONTAL 1.000E+00,  
 2.000E+00;VERTICAL 1.000E+00,  
 2.000E+00;;HISTORY:CURRENT:SEARCH:  
 SELECT1:TRACE 1;;HISTORY:CURRENT:  
 SEARCH:SELECT1:WAVE:TRANGE 1.000E+00,  
 2.000E+00;;HISTORY:CURRENT:SEARCH:  
 SELECT1:WINDOW MAIN;;HISTORY:CURRENT:  
 SEARCH:SELECT2:CONDITION IN;  
 MODE PARAMETER;PARAMETER:  
 CATEGORY MEASURE;MEASURE:  
 CALCULATION1 1.000E+00,2.000E+00;;  
 HISTORY:CURRENT:SEARCH:SELECT2:  
 RECTANGLE:HORIZONTAL 1.000E+00,  
 2.000E+00;VERTICAL 1.000E+00,  
 2.000E+00;;HISTORY:CURRENT:SEARCH:  
 SELECT2:TRACE 1;;HISTORY:CURRENT:  
 SEARCH:SELECT2:WAVE:TRANGE 1.000E+00,  
 2.000E+00;;HISTORY:CURRENT:SEARCH:  
 SELECT2:WINDOW MAIN;;HISTORY:CURRENT:  
 SEARCH;SELECT3:CONDITION IN;  
 MODE PARAMETER;PARAMETER:  
 CATEGORY MEASURE;MEASURE:  
 CALCULATION1 1.000E+00,2.000E+00.....

### **:HISTory[:CURRENT]:[SEARCH]:ABORT**

**Function** Aborts the history search.

**Syntax** :HISTory[:CURRENT]:[SEARCH]:ABORT

**Example** :HISTORY:CURRENT:SEARCH:ABORT

### **:HISTory[:CURRENT]:[SEARCH]:EXECute**

**Function** Executes the history search. This is an overlap command.

**Syntax** :HISTory[:CURRENT]:[SEARCH]:EXECute

**Example** :HISTORY:CURRENT:SEARCH:EXECUTE

### **:HISTory[:CURRENT]:[SEARCH]:LOGic**

**Function** Sets the history search logic or queries the current setting.

**Syntax** :HISTory[:CURRENT]:[SEARCH]:LOGic {AND|OR}

:HISTory[:CURRENT]:[SEARCH]:LOGic?

**Example** :HISTORY:CURRENT:SEARCH:LOGIC AND  
 :HISTORY:CURRENT:SEARCH:LOGIC?  
 -> :HISTORY:CURRENT:SEARCH:LOGIC AND

### **:HISTory[:CURRENT]:[SEARCH]:RESet**

**Function** Resets the search conditions of the history search.

**Syntax** :HISTory[:CURRENT]:[SEARCH]:RESet

**Example** :HISTORY:CURRENT:SEARCH:RESET

### **:HISTory[:CURRENT]:[SEARCH]:SElect<x>?**

**Function** Queries all settings related to the history search condition.

**Syntax** :HISTory[:CURRENT]:[SEARCH]:SElect<x>?  
 <x> = 1 to 4

**Example** :HISTORY:CURRENT:SEARCH:SELECT1?  
 -> :HISTORY:CURRENT:SEARCH:SELECT1:  
 CONDITION IN;MODE PARAMETER;PARAMETER:  
 CATEGORY MEASURE;MEASURE:  
 CALCULATION1 1.000E+00,2.000E+00;;  
 HISTORY:CURRENT:SEARCH:SELECT1:  
 RECTANGLE:HORIZONTAL 1.000E+00,  
 2.000E+00;VERTICAL 1.000E+00,  
 2.000E+00;;HISTORY:CURRENT:SEARCH:  
 SELECT1:TRACE 1;;HISTORY:CURRENT:  
 SEARCH:SELECT1:WAVE:TRANGE 1.000E+00,  
 2.000E+00;;HISTORY:CURRENT:SEARCH:  
 SELECT1:WINDOW MAIN

### **:HISTory[:CURRENT]:[SEARCH]:SElect<x>:CONDition**

**Function** Sets determination criteria of the history search condition or queries the current setting.

**Syntax** :HISTory[:CURRENT]:[SEARCH]:SElect<x>:  
 CONDition {DONTcare|IN|OUT}  
 :HISTory[:CURRENT]:[SEARCH]:SElect<x>:  
 CONDition?  
 <x> = 1 to 4

**Example** :HISTORY:CURRENT:SEARCH:SELECT1:  
 CONDITION IN  
 :HISTORY:CURRENT:SEARCH:SELECT1:  
 CONDITION? -> :HISTORY:CURRENT:SEARCH:  
 SELECT1:CONDITION IN

### **:HISTory[:CURRENT]:[SEARCH]:SElect<x>:MODE**

**Function** Sets the mode of the history search condition or queries the current setting.

**Syntax** :HISTory[:CURRENT]:[SEARCH]:SElect<x>:  
 MODE {PARAMeter|POLYgon|RECTangle|WAVE}  
 :HISTory[:CURRENT]:[SEARCH]:SElect<x>:  
 MODE?  
 <x> = 1 to 4

**Example** :HISTORY:CURRENT:SEARCH:SELECT1:  
 MODE WAVE  
 :HISTORY:CURRENT:SEARCH:SELECT1:MODE?  
 -> :HISTORY:CURRENT:SEARCH:SELECT1:  
 MODE WAVE

**:HISTory[:CURRENT][:SEARCH]:SElect<x>:  
PARAmeter?**

**Function** Queries all settings related to the parameter of the history search condition.

**Syntax** :HISTory[:CURRENT][:SEARCH]:SElect<x>:  
PARAmeter?  
<x> = 1 to 4

**Example** :HISTORY:CURRENT:SEARCH:SELECT1:  
PARAMETER? -> :HISTORY:CURRENT:SEARCH:  
SELECT1:PARAMETER:CATEGORY MEASURE;  
MEASURE:CALCULATION1 2.000E+00,  
1.000E+00

**:HISTory[:CURRENT][:SEARCH]:SElect<x>:  
PARAmeter:CATegory**

**Function** Sets the parameter category or queries the current setting.

**Syntax** :HISTory[:CURRENT][:SEARCH]:SElect<x>:  
PARAmeter:CATegory {FFT|MEASure|XY}  
:HISTory[:CURRENT][:SEARCH]:SElect<x>:  
PARAmeter:CATegory?  
<x> = 1 to 4

**Example** :HISTORY:CURRENT:SEARCH:SELECT1:  
PARAMETER:CATEGORY MEASURE  
:HISTORY:CURRENT:SEARCH:SELECT1:  
PARAMETER:CATEGORY? -> :HISTORY:  
CURRENT:SEARCH:SELECT1:  
PARAMETER:CATEGORY MEASURE

**:HISTory[:CURRENT][:SEARCH]:SElect<x>:  
PARAmeter:FFT<x>?**

**Function** Queries all settings related to the FFT search.

**Syntax** :HISTory[:CURRENT][:SEARCH]:SElect<x>:  
PARAmeter:FFT<x>?  
<x> of SElect<x> = 1 to 4  
<x> of FFT<x> = 1 or 2

**Example** :HISTORY:CURRENT:SEARCH:SELECT1:  
PARAMETER:FFT1? -> :HISTORY:CURRENT:  
SEARCH:SELECT1:PARAMETER:FFT1:  
CALCULATION1 2.000E+00,1.000E+00

**:HISTory[:CURRENT][:SEARCH]:SElect<x>:  
PARAmeter:FFT<x>:CALCulation<x>**

**Function** Sets the upper and lower limits of the calculation item of the FFT search or queries the current setting.

**Syntax** :HISTory[:CURRENT][:SEARCH]:SElect<x>:  
PARAmeter:FFT<x>:CALCulation<x>  
{<NRf>,<NRf>}

**Example** :HISTory[:CURRENT][:SEARCH]:SElect<x>:  
PARAmeter:FFT<x>:CALCulation<x>?  
<x> of SElect<x> = 1 to 4  
<x> of FFT<x> = 1 or 2  
<x> of CALCulation<x> = 1 to 4  
<NRf> = See the DL9000 User's Manual.

**Example** :HISTORY:CURRENT:SEARCH:SELECT1:  
PARAMETER:FFT1:CALCULATION1 1,2  
:HISTORY:CURRENT:SEARCH:SELECT1:  
PARAMETER:FFT1:CALCULATION1?  
-> :HISTORY:CURRENT:SEARCH:SELECT1:  
PARAMETER:FFT1:CALCULATION1 2.000E+00,  
1.000E+00

**:HISTory[:CURRENT][:SEARCH]:SElect<x>:  
PARAmeter:FFT<x>:PEAK?**

**Function** Queries all settings related to the peak value of the FFT search.

**Syntax** :HISTory[:CURRENT][:SEARCH]:SElect<x>:  
PARAmeter:FFT<x>:PEAK?  
<x> of SElect<x> = 1 to 4  
<x> of FFT<x> = 1 or 2

**Example** :HISTORY:CURRENT:SEARCH:SELECT1:  
PARAMETER:FFT1:PEAK? -> :HISTORY:  
CURRENT:SEARCH:SELECT1:PARAMETER:FFT1:  
PEAK:DV 2.000E+00,1.000E+00

## 5.14 HISTory Group

### **:HISTory[:CURRENT][:SEARCH]:SElect<x>:**

#### **PARAMeter:FFT<x>:PEAK:DFREquency**

**Function** Sets the upper and lower limits between the peak frequencies of the FFT search or queries the current setting.

**Syntax** :HISTory[:CURRENT][:SEARCH]:SElect<x>:  
PARAMeter:FFT<x>:PEAK:DFREquency {<Frequency>,<Frequency>}  
:HISTory[:CURRENT][:SEARCH]:SElect<x>:  
PARAMeter:FFT<x>:PEAK:DFREquency?  
<x> of SElect<x> = 1 to 4  
<x> of FFT<x> = 1 or 2  
<Frequency> = See the DL9000 User's Manual

**Example** :HISTORY:CURRENT:SEARCH:SELECT1:  
PARAMETER:FFT1:PEAK:DFREQUENCY 1,10  
:HISTORY:CURRENT:SEARCH:SELECT1:  
PARAMETER:FFT1:PEAK:DFREQUENCY?  
-> :HISTORY:CURRENT:SEARCH:SELECT1:  
PARAMETER:FFT1:PEAK:  
DFREQUENCY 10.000E+00,1.000E+00

### **:HISTory[:CURRENT][:SEARCH]:SElect<x>:**

#### **PARAMeter:FFT<x>:PEAK:DV**

**Function** Sets the upper and lower limits between the peak voltages of the FFT search or queries the current setting.

**Syntax** :HISTory[:CURRENT][:SEARCH]:SElect<x>:  
PARAMeter:FFT<x>:PEAK:DV {<NRf>,<NRf>}  
:HISTory[:CURRENT][:SEARCH]:SElect<x>:  
PARAMeter:FFT<x>:PEAK:DV?  
<x> of SElect<x> = 1 to 4  
<x> of FFT<x> = 1 or 2  
<NRf> = -4 to 4 (div)

**Example** :HISTORY:CURRENT:SEARCH:SELECT1:  
PARAMETER:FFT1:PEAK:DV 1,2  
:HISTORY:CURRENT:SEARCH:SELECT1:  
PARAMETER:FFT1:PEAK:DV? -> :HISTORY:  
CURRENT:SEARCH:SELECT1:PARAMETER:FFT1:  
PEAK:DV 2.000E+00,1.000E+00

### **:HISTory[:CURRENT][:SEARCH]:SElect<x>:**

#### **PARAMeter:FFT<x>:PEAK:FREquency<x>**

**Function** Sets the upper and lower limits of the peak frequency of the FFT search or queries the current setting.

**Syntax** :HISTory[:CURRENT][:SEARCH]:SElect<x>:  
PARAMeter:FFT<x>:PEAK:FREquency<x>  
{<Frequency>,<Frequency>}  
:HISTory[:CURRENT][:SEARCH]:SElect<x>:  
PARAMeter:FFT<x>:PEAK:FREquency<x>?  
<x> of SElect<x> = 1 to 4  
<x> of FFT<x> = 1 or 2  
<x> of FREquency<x> = 1 or 2  
<Frequency> = See the DL9000 User's Manual

**Example** :HISTORY:CURRENT:SEARCH:SELECT1:  
PARAMETER:FFT1:PEAK:FREQUENCY1 1,2  
:HISTORY:CURRENT:SEARCH:SELECT1:  
PARAMETER:FFT1:PEAK:FREQUENCY1?  
-> :HISTORY:CURRENT:SEARCH:SELECT1:  
PARAMETER:FFT1:PEAK:  
FREQUENCY1 2.000E+00,1.000E+00

### **:HISTory[:CURRENT][:SEARCH]:SElect<x>:**

#### **PARAMeter:FFT<x>:PEAK:V<x>**

**Function** Sets the upper and lower limits of the peak voltage of the FFT search or queries the current setting.

**Syntax** :HISTory[:CURRENT][:SEARCH]:SElect<x>:  
PARAMeter:FFT<x>:PEAK:V<x>  
{<NRf>,<NRf>}  
:HISTory[:CURRENT][:SEARCH]:SElect<x>:  
PARAMeter:FFT<x>:PEAK:V<x>?  
<x> of SElect<x> = 1 to 4  
<x> of FFT<x> = 1 or 2  
<x> of V<x> = 1 or 2  
<NRf> = -4 to 4 (div)

**Example** :HISTORY:CURRENT:SEARCH:SELECT1:  
PARAMETER:FFT1:PEAK:V1 1,2  
:HISTORY:CURRENT:SEARCH:SELECT1:  
PARAMETER:FFT1:PEAK:V1? -> :HISTORY:  
CURRENT:SEARCH:SELECT1:PARAMETER:FFT1:  
PEAK:V1 2.000E+00,1.000E+00

### **:HISTory[:CURRENT][:SEARCH]:SElect<x>:**

#### **PARAMeter:MEASure?**

**Function** Queries all settings related to the search using automated measurement of waveform parameters (measure search).

**Syntax** :HISTory[:CURRENT][:SEARCH]:SElect<x>:  
PARAMeter:MEASure?  
<x> = 1 to 4

**Example** :HISTORY:CURRENT:SEARCH:SELECT1:  
PARAMETER:MEASURE? -> :HISTORY:CURRENT:  
SEARCH:SELECT1:PARAMETER:MEASURE:  
CALCULATION1 1.000E+00,2.000E+00

**:HISToRY[:CURRent][:SEARCh]:SElect<x>:  
PARAmeter:MEASure:CALCulation<x>**

**Function** Sets the upper and lower limits of the calculation item of the measure search or queries the current setting.

**Syntax** :HISToRY[:CURRent][:SEARCh]:SElect<x>:  
PARAmeter:MEASure:CALCulation<x>  
{<NRf>,<NRf>}  
:HISToRY[:CURRent][:SEARCh]:SElect<x>:  
PARAmeter:MEASure:CALCulation<x>?  
<x> of SElect<x> = 1 to 4  
<x> of CALCulation<x> = 1 to 4  
<NRf> = -4 to 4 (div)

**Example** :HISToRY:CURRent:SEARCh:SElect1:  
PARAmeter:MEASure:CALCulation1 1,2  
:HISToRY:CURRent:SEARCh:SElect1:  
PARAmeter:MEASure:CALCulation1?  
-> :HISToRY:CURRent:SEARCh:SElect1:  
PARAmeter:MEASure:  
CALCulation1 2.000E+00,1.000E+00

**:HISToRY[:CURRent][:SEARCh]:SElect<x>:  
PARAmeter:MEASure:TRACe<x>?**

**Function** Queries all settings related to the trace of the measure search.

**Syntax** :HISToRY[:CURRent][:SEARCh]:SElect<x>:  
PARAmeter:MEASure:TRACe<x>?  
<x> of SElect<x> = 1 to 4  
<x> of TRACe<x> = 1 to 8

**Example** :HISToRY:CURRent:SEARCh:SElect1:  
PARAmeter:MEASure:TRACe1? -> :HISToRY:  
CURRent:SEARCh:SElect1:  
PARAmeter:MEASure:TRACe1:AREa1:TYPE:  
BURST 2.000E+00,1.000E+00

**:HISToRY[:CURRent][:SEARCh]:SElect<x>:  
PARAmeter:MEASure:TRACe<x>:AREa<x>?**

**Function** Queries all settings related to the area of the measure search.

**Syntax** :HISToRY[:CURRent][:SEARCh]:SElect<x>:  
PARAmeter:MEASure:TRACe<x>:AREa<x>?  
<x> of SElect<x> = 1 to 4  
<x> of TRACe<x> = 1 to 8  
<x> of AREa<x> = 1 or 2

**Example** :HISToRY:CURRent:SEARCh:SElect1:  
PARAmeter:MEASure:TRACe1:AREa1?  
-> :HISToRY:CURRent:SEARCh:SElect1:  
PARAmeter:MEASure:TRACe1:AREa1:TYPE:  
BURST 2.000E+00,1.000E+00SElect<x>:  
PARAmeter:MEASure:TRACe<x>:AREa<x>:  
TYPE?

**:HISToRY[:CURRent][:SEARCh]:SElect<x>:  
PARAmeter:MEASure:TRACe<x>:AREa<x>:  
TYPE?**

**Function** Queries all settings related to the waveform parameters of the measure search.

**Syntax** :HISToRY[:CURRent][:SEARCh]:SElect<x>:  
PARAmeter:MEASure:TRACe<x>:AREa<x>:  
TYPE?  
<x> of SElect<x> = 1 to 4  
<x> of TRACe<x> = 1 to 8  
<x> of AREa<x> = 1 or 2

**Example** :HISToRY:CURRent:SEARCh:SElect1:  
PARAmeter:MEASure:TRACe1:AREa1:TYPE?  
-> :HISToRY:CURRent:SEARCh:SElect1:  
PARAmeter:MEASure:TRACe1:AREa1:TYPE:  
BURST 2.000E+00,1.000E+00

## 5.14 HISTory Group

**:HISTory[:CURRENT][:SEARCH]:SElect<x>:**

**PARAMeter:MEASure:TRACe<x>:AREA<x>:**

**TYPE:<Parameter>**

**Function** Sets the upper and lower limits of the waveform parameter of the measure search or queries the current setting.

**Syntax** :HISTory[:CURRENT][:SEARCH]:SElect<x>:  
PARAMeter:MEASure:TRACe<x>:AREA<x>:  
TYPE:<Parameter> {( <Nrf>, <Nrf> ) |  
( <Voltage>, <Voltage> ) |  
( <Current>, <Current> ) | ( <Time>, <Time> ) |  
( <Frequency>, <Frequency> ) }  
:HISTory[:CURRENT][:SEARCH]:SElect<x>:  
PARAMeter:MEASure:TRACe<x>:AREA<x>:  
TYPE:<Parameter>?  
<x> of SElect<x> = 1 to 4  
<x> of TRACe<x> = 1 to 8  
<x> of AREA<x> = 1 or 2  
<Parameter> = {BURSt|CMEan|COUNT|CRMS|  
CSSDeviation|DElay|DT|DUTYcycle|FALL|  
FREQuency|HIGH|HILow|LOW|MAXimum|MEAN|  
MINimum|NOVershoot|NWIDTH|PERFrequency|  
PERiod|POVershoot|PTOPeak|PWIDTH|RISE|  
RMS|SDEViation|TYCInteg|TYINteg|V1|V2|  
<Nrf>, <Voltage>, <Current>, <Time>, and  
<Frequency> = See the DL9000 User's Manual.

**Example** (The following is an example for the maximum value of trace 1 and area 1.)

```
:HISTORY:CURRENT:SEARCH:SELECT1:
PARAMETER:MEASURE:TRACE1:AREA1:TYPE:
MAXIMUM 0,1
:HISTORY:CURRENT:SEARCH:SELECT1:
PARAMETER:MEASURE:TRACE1:AREA1:TYPE:
MAXIMUM? -> :HISTORY:CURRENT:SEARCH:
SELECT1:PARAMETER:MEASURE:TRACE1:AREA1:
TYPE:MAXIMUM 1.000E+00,0.000E+00
```

**:HISTory[:CURRENT][:SEARCH]:SElect<x>:**

**PARAMeter:XY<x>?**

**Function** Queries all settings related to the XY search.

**Syntax** :HISTory[:CURRENT][:SEARCH]:SElect<x>:  
PARAMeter:XY<x>?  
<x> of SElect<x> = 1 to 4  
<x> of XY<x> = 1 or 2

**Example** :HISTORY:CURRENT:SEARCH:SELECT1:  
PARAMETER:XY1? -> :HISTORY:CURRENT:  
SEARCH:SELECT1:PARAMETER:XY1:  
XYINTEG 2.000E+00,1.000E+00

**:HISTory[:CURRENT][:SEARCH]:SElect<x>:**

**PARAMeter:XY<x>:XYINteg**

**Function** Sets the upper and lower limits integral value of the XY search or queries the current setting.

**Syntax** :HISTory[:CURRENT][:SEARCH]:SElect<x>:  
PARAMeter:XY<x>:XYINteg {<Nrf>,<Nrf>}  
:HISTory[:CURRENT][:SEARCH]:SElect<x>:  
PARAMeter:XY<x>:XYINteg?  
<x> of SElect<x> = 1 to 4  
<x> of XY<x> = 1 or 2  
<Nrf> = -4 to 4 (div)

**Example** :HISTORY:CURRENT:SEARCH:SELECT1:  
PARAMETER:XY1:XYINTEG 1,2  
:HISTORY:CURRENT:SEARCH:SELECT1:  
PARAMETER:XY1:XYINTEG? -> :HISTORY:  
CURRENT:SEARCH:SELECT1:  
PARAMETER:XY1:  
XYINTEG 2.000E+00,1.000E+00

**:HISTory[:CURRENT][:SEARCH]:SElect<x>:  
RECTangle?**

**Function** Queries all settings related to the rectangle search.

**Syntax** :HISTory[:CURRENT][:SEARCH]:SElect<x>:  
RECTangle?  
<x> = 1 to 4

**Example** :HISTORY:CURRENT:SEARCH:SELECT1:  
RECTANGLE? -> :HISTORY:CURRENT:SEARCH:  
SELECT1:RECTANGLE:HORIZONTAL 2.000E+00,  
1.000E+00;VERTICAL 2.000E+00,1.000E+00

**:HISTory[:CURRENT][:SEARCH]:SElect<x>:  
RECTangle:HORizontal**

**Function** Sets the horizontal position of the rectangle used in the rectangle search or queries the current setting.

**Syntax** :HISTory[:CURRENT][:SEARCH]:SElect<x>:  
RECTangle:HORizontal {<Nrf>,<Nrf>}  
:HISTory[:CURRENT][:SEARCH]:SElect<x>:  
RECTangle:HORizontal?  
<x> = 1 to 4  
<Nrf> = -5 to 5 (div)

**Example** :HISTORY:CURRENT:SEARCH:SELECT1:  
RECTANGLE:HORIZONTAL 1,2  
:HISTORY:CURRENT:SEARCH:SELECT1:  
RECTANGLE:HORIZONTAL? -> :HISTORY:  
CURRENT:SEARCH:SELECT1:RECTANGLE:  
HORIZONTAL 2.000E+00,1.000E+00

**:HISTory[:CURRENT][:SEARCH]:SElect<x>:  
RECTangle:VERTical**

**Function** Sets the vertical position of the rectangle used in the rectangle search or queries the current setting.

**Syntax** :HISTory[:CURRENT][:SEARCH]:SElect<x>:  
RECTangle:VERTical {<NRf>,<NRf>}  
:HISTory[:CURRENT][:SEARCH]:SElect<x>:  
RECTangle:VERTical?  
<x> = 1 to 4  
<NRf> = -4 to 4 (div)

**Example** :HISTORY:CURRENT:SEARCH:SELECT1:  
RECTANGLE:VERTICAL 1,2  
:HISTORY:CURRENT:SEARCH:SELECT1:  
RECTANGLE:VERTICAL? -> :HISTORY:  
CURRENT:SEARCH:SELECT1:  
RECTANGLE:VERTICAL 2.000E+00,1.000E+00

**:HISTory[:CURRENT][:SEARCH]:SElect<x>:  
TRACe**

**Function** Sets the source trace of the history search or queries the current setting.

**Syntax** :HISTory[:CURRENT][:SEARCH]:SElect<x>:  
TRACe {<NRf>}  
:HISTory[:CURRENT][:SEARCH]:SElect<x>:  
TRACe?  
<x> = 1 to 4  
<NRf> = 1 to 8

**Example** :HISTORY:CURRENT:SEARCH:SELECT1:TRACE 1  
:HISTORY:CURRENT:SEARCH:SELECT1:TRACE?  
-> :HISTORY:CURRENT:SEARCH:SELECT1:  
TRACE 1

**Description** This command is invalid when Window is XY.

**:HISTory[:CURRENT][:SEARCH]:SElect<x>:  
WAVE?**

**Function** Queries all settings related to the search in the waveform zone (wave search).

**Syntax** :HISTory[:CURRENT][:SEARCH]:SElect<x>:  
WAVE?  
<x> = 1 to 4

**Example** :HISTORY:CURRENT:SEARCH:SELECT1:WAVE?  
-> :HISTORY:CURRENT:SEARCH:SELECT1:  
WAVE:TRANGE 2.000E+00,1.000E+00

**:HISTory[:CURRENT][:SEARCH]:SElect<x>:  
WAVE:EDIT<x>:EXIT**

**Function** Exits the zone edit menu of the wave search.

**Syntax** :HISTory[:CURRENT][:SEARCH]:SElect<x>:  
WAVE:EDIT<x>:EXIT {<NRf>|QUIT}  
<x> of SElect<x> = 1 to 4  
<x> of EDIT<x> = 1 to 13  
<NRf> = 1 to 4 (internal memories)

**Example** :HISTORY:CURRENT:SEARCH:SELECT1:WAVE:  
EDIT1:EXIT 1

**Description** • An error occurs if this command is issued when the zone is not being edited.

- Saves the zone waveform to the internal memory specified by <NRf>.

**:HISTory[:CURRENT][:SEARCH]:SElect<x>:  
WAVE:EDIT<x>:PART**

**Function** Sets the editing of the portion of the zone of the wave search.

**Syntax** :HISTory[:CURRENT][:SEARCH]:SElect<x>:  
WAVE:EDIT<x>:PART {<NRf>,<NRf>,<NRf>,<NRf>}  
<x> of SElect<x> = 1 to 4

<x> of EDIT<x> = 1 to 13 (1 to 8 are traces. 9 to 12 are internal memories. 13 is the zone waveform that is currently displayed)

<NRf> = -5 to 5 (div: partial editing time axis cursor 1 and 2)  
-8 to 8 (div: up and down) (in the order partial editing time axis cursor 1, 2, ↑, and ↓)

**Example** :HISTORY:CURRENT:SEARCH:SELECT1:WAVE:  
EDIT1:PART 1,2,3,4

**Description** The amount of movement upward or downward is a relative value with respect to the zone waveform that is currently displayed.

**:HISTory[:CURRENT][:SEARCH]:SElect<x>:  
WAVE:EDIT<x>:WHOLE**

**Function** Sets the editing of the entire zone of the wave search.

**Syntax** :HISTory[:CURRENT][:SEARCH]:SElect<x>:  
WAVE:EDIT<x>:  
WHOLE {<NRf>,<NRf>,<NRf>,<NRf>}  
<x> of SElect<x> = 1 to 4

<x> of EDIT<x> = 1 to 13 (1 to 8 are traces. 9 to 12 are internal memories. 13 is the zone waveform that is currently displayed)

<NRf> = 0 to 5 (div: left and right)  
0 to 8 (div: up and down)  
(In the order ←, →, ↑, and ↓)

**Example** :HISTORY:CURRENT:SEARCH:SELECT1:WAVE:  
EDIT1:WHOLE 1,2,3,4

**Description** The amount of movement is a relative value with respect to the reference waveform.

## 5.14 HISTory Group

### **:HISTory[:CURRENT][:SEARCH]:SElect<x>: WAVE:TRANge**

**Function** Sets the range over which to perform the wave search or queries the current setting.

**Syntax** :HISTory[:CURRENT][:SEARCH]:SElect<x>:  
WAVE:TRANge {<NRf>,<NRf>}  
:HISTory[:CURRENT][:SEARCH]:SElect<x>:  
WAVE:TRANge?  
<x> = 1 to 4  
<NRf> = -5 to 5 (div)

**Example** :HISTORY:CURRENT:SEARCH:SELECT1:WAVE:  
TRANGE 1,2  
:HISTORY:CURRENT:SEARCH:SELECT1:WAVE:  
TRANGE? -> :HISTORY:CURRENT:SEARCH:  
SELECT1:WAVE:TRANGE 2.000E+00,1.000E+00

### **:HISTory[:CURRENT][:SEARCH]:SElect<x>: WINDOW**

**Function** Sets the source window of the history search or queries the current setting.

**Syntax** :HISTory[:CURRENT][:SEARCH]:SElect<x>:  
WINDOW {MAIN|XY1|XY2|Z1|Z2}  
:HISTory[:CURRENT][:SEARCH]:SElect<x>:  
WINDOW?  
<x> = 1 to 4

**Example** :HISTORY:CURRENT:SEARCH:SELECT1:  
WINDOW MAIN  
:HISTORY:CURRENT:SEARCH:SELECT1:WINDOW?  
-> :HISTORY:CURRENT:SEARCH:SELECT1:  
WINDOW MAIN

Description XY1 and XY2 are valid only for a rectangular polygon.

### **:HISTory[:CURRENT]:TIME?**

**Function** Queries the time of the source record number of the history waveform.

**Syntax** :HISTory[:CURRENT]:TIME? {<NRf>|  
MINimum}

**Example** :HISTORY:CURRENT:TIME? -1 -> :HISTORY:  
CURRENT:TIME "-1 10:20:30.10"

Description Specifying MINimum sets the record to the minimum record number.

### **:HISTory:REFeRence<x>?**

**Function** Queries all settings related to the history function of the reference.

**Syntax** :HISTory:REFeRence<x>?  
<x> = 1 to 4

**Example** :HISTORY:REFERENCE1? -> HISTORY:  
REFERENCE1:DMODE ACOLOR;  
MODE AVERAGE;RECORD 1;REPLAY:SPEED 1

### **:HISTory:REFeRence<x>:DMODE (Display Mode)**

**Function** Sets the display mode of the history waveform of the reference or queries the current setting.

**Syntax** :HISTory:REFeRence<x>:DMODE {ACOLor|  
AHTone|AINTensity|ONE}  
:HISTory:REFeRence<x>:DMODE?  
<x> = 1 to 4

**Example** :HISTORY:REFERENCE1:DMODE ACOLOR  
:HISTORY:REFERENCE1:DMODE? -> :HISTORY:  
REFERENCE1:DMODE ACOLOR

### **:HISTory:REFeRence<x>:MODE**

**Function** Sets the highlight display mode of the history waveform of the reference or queries the current setting.

**Syntax** :HISTory:REFeRence<x>:MODE {AVERAge|  
RECOrd}  
:HISTory:REFeRence<x>:MODE?  
<x> = 1 to 4

**Example** :HISTORY:REFERENCE1:MODE AVERAGE  
:HISTORY:REFERENCE1:MODE? -> :HISTORY:  
REFERENCE1:MODE AVERAGE

### **:HISTory:REFeRence<x>:RECOrd**

**Function** Sets the source record of the history waveform of the reference or queries the current setting.

**Syntax** :HISTory:REFeRence<x>:RECOrd {<NRf>|  
MINimum}  
:HISTory:REFeRence<x>:RECOrd?  
<x> = 1 to 4  
<NRf> = See the DL9000 User's Manual.

**Example** :HISTORY:REFERENCE1:RECORD 1  
:HISTORY:REFERENCE1:RECORD?  
-> :HISTORY:REFERENCE1:RECORD 1

### **:HISTory:REFeRence<x>:RECOrd? MINimum**

**Function** Queries the minimum record number of the history waveform of the reference.

**Syntax** :HISTory:REFeRence<x>:RECOrd? {MINimum}  
<x> = 1 to 4

**Example** :HISTORY:REFERENCE1:RECORD? MINIMUM  
-> :HISTORY:REFERENCE1:RECORD -1

Description Specifying MINimum sets the record to the minimum record number.

### **:HISTory:REFeRence<x>:REPLAy?**

**Function** Queries all settings related to the replay function of the history function of the reference.

**Syntax** :HISTory:REFeRence<x>:REPLAy?  
<x> = 1 to 4

**Example** :HISTORY:REFERENCE1:REPLAY?  
-> :HISTORY:REFERENCE1:REPLAY:SPEED 1



**: HISToRY: REFeRence<x>: REPLay: JUMP**

Function Jumps to the specified record number of the history waveform of the reference.

Syntax :HISToRY:REFeRence<x>:REPLay:JUMP  
{MAXimum|MINimum}  
<x> = 1 to 4

Example :HISTORY:REFERENCE1:REPLAY:JUMP MAXIMUM

**: HISToRY: REFeRence<x>: REPLay: SPEEd**

Function Sets the replay speed of the history waveform of the reference or queries the current setting.

Syntax :HISToRY:REFeRence<x>:REPLay:SPEEd  
{<NRf>|PER3|PER10|PER30|PER60}  
:HISToRY:REFeRence<x>:REPLay:SPEEd?  
<x> = 1 to 4  
<NRf> = 1,3,10

Example :HISTORY:REFERENCE1:REPLAY:SPEED 1  
:HISTORY:REFERENCE1:REPLAY:SPEED?  
-> :HISTORY:REFERENCE1:REPLAY:SPEED 1

**: HISToRY: REFeRence<x>: REPLay: START**

Function Starts the replay of the history waveform of the reference.

Syntax :HISToRY:REFeRence<x>:REPLay:START  
{MAXimum|MINimum}  
<x> = 1 to 4

Example :HISTORY:REFERENCE1:REPLAY:START  
MAXIMUM

**: HISToRY: REFeRence<x>: REPLay: STOP**

Function Stops the replay of the history waveform of the reference.

Syntax :HISToRY:REFeRence<x>:REPLay:STOP  
<x> = 1 to 4

Example :HISTORY:REFERENCE1:REPLAY:STOP

**: HISToRY: REFeRence<x>: TIME?**

Function Queries the time of the source record number of the reference waveform.

Syntax :HISToRY:REFeRence<x>:TIME? {<NRf>|  
MINimum}  
<x> = 1 to 4

Example :HISTORY:REFERENCE1:TIME? -1  
-> :HISTORY:REFERENCE1:  
TIME "-1 10:20:30.10"

Description Specifying MINimum sets the record to the minimum record number.

---

## 5.15 IMAGE Group

### **: IMAGE?**

Function Queries all settings related to the output of screen image data.

Syntax :IMAGE?

Example :IMAGE? -> :IMAGE:FORMAT BMP;TONE COLOR

### **: IMAGE: FORMat**

Function Sets the output format of the screen image data or queries the current setting.

Syntax :IMAGE:FORMat {BMP|JPEG|PNG}

:IMAGE:FORMat?

Example :IMAGE:FORMAT BMP

:IMAGE:FORMAT? -> :IMAGE:FORMAT BMP

### **: IMAGE: SEND?**

Function Queries the screen image data.

Syntax :IMAGE:SEND?

Example :IMAGE:SEND? -> #6 (number of bytes, 6 digits) (data byte sequence) (block data)

Description For details on <Block data>, see page 4-6.

### **: IMAGE: TONE**

Function Sets the tone of the screen image data or queries the current setting.

Syntax :IMAGE:TONE {COLor|GRAY|OFF|REVerse}

:IMAGE:TONE?

Example :IMAGE:TONE COLOR

:IMAGE:TONE? -> :IMAGE:TONE COLOR

Description If “:IMAGE:FORMat JPEG” is specified, OFF cannot be selected.

---

## 5.16 INITialize Group

### **:INITialize:EXECute**

Function Execute the initialization.

Syntax :INITialize:EXECute

Example :INITIALIZE:EXECUTE

### **:INITialize:UNDO**

Function Cancels the initialization that was executed.

Syntax :INITialize:UNDO

Example :INITIALIZE:UNDO

## 5.17 MATH Group

### **:MATH<x>?**

**Function** Queries all settings related to the computation.

**Syntax** :MATH<x>?  
<x> = 1 to 8

**Example** :MATH1? -> :MATH1:SELECT MATH;  
DISPLAY 1;ECOUNT:HYSTERESIS 1.000E+00;  
POLARITY RISE;:MATH1:FILTER:DELAY:  
TIME 0.000E+00;:MATH1:FILTER:IIR:  
FORDER 2;HIPASS:COFF 10.00E+06;:MATH1:  
FILTER:IIR:LOWPASS:COFF 10.00E+00;:  
MATH1:FILTER:MAVG:WEIGHT 2;:MATH1:  
FILTER:RESCALING:AVALUE 1.000E+00;  
BVALUE 0.000E+00;:MATH1:FILTER:  
TYPE THROUGH;:MATH1:INTEGRAL:PSCALING:  
AVALUE 1.000E+00;BVALUE 0.000E+00;:  
MATH1:INTEGRAL:RESCALING:  
AVALUE 1.000E+00;BVALUE 0.000E+00;:  
MATH1:INVERT 0;IPOINT:  
POSITION 0.000E+00;:MATH1:LABEL:  
DEFINE "MATH1";MODE 0;:MATH1:MINUS:  
PSCALING1:AVALUE 1.000E+00;  
BVALUE 0.000E+00;:MATH1:MINUS:  
PSCALING2:AVALUE 1.000E+00;  
BVALUE 0.000E+00;:MATH1:MINUS:  
RESCALING:AVALUE 1.000E+00;  
BVALUE 0.000E+00;:MATH1:MULTIPLE:  
PSCALING1:AVALUE 1.000E+00;  
BVALUE 0.000E+00;:MATH1:MULTIPLE:  
PSCALING2:AVALUE 1.000E+00;  
BVALUE 0.000E+00;:MATH1:MULTIPLE:  
RESCALING:AVALUE 1.000E+00;  
BVALUE 0.000E+00;:MATH1:  
OPERATION FILTER,1;PLUS:PSCALING1:  
AVALUE 1.000E+00;BVALUE .....

### **:MATH<x>:DISPlay**

**Function** Turns ON/OFF the computed waveform or queries the current setting.

**Syntax** :MATH<x>:DISPlay {<Boolean>}  
:MATH<x>:DISPlay?

<x> = 1 to 8

**Example** :MATH1:DISPLAY ON  
:MATH1:DISPLAY? -> :MATH1:DISPLAY 1

### **:MATH<x>:ECOUNT?**

**Function** Queries all settings related to the edge count computation.

**Syntax** :MATH<x>:ECOUNT?  
<x> = 1 to 8

**Example** :MATH1:ECOUNT? -> :MATH1:ECOUNT:  
HYSTERESIS 1.000E+00;POLARITY RISE

### **:MATH<x>:ECOUNT:HYSTERESIS**

**Function** Sets the hysteresis of the edge detection level of the edge count computation or queries the current setting.

**Syntax** :MATH<x>:ECOUNT:HYSTERESIS {<NRf>}  
:MATH<x>:ECOUNT:HYSTERESIS?  
<x> = 1 to 8  
<NRf> = 0 to 4 (div)

**Example** :MATH1:ECOUNT:HYSTERESIS 1  
:MATH1:ECOUNT:HYSTERESIS? -> :MATH1:  
ECOUNT:HYSTERESIS 1.000E+00

### **:MATH<x>:ECOUNT:POLarity**

**Function** Sets the edge detection polarity of the edge count computation or queries the current setting.

**Syntax** :MATH<x>:ECOUNT:POLarity {FALL|RISE}  
:MATH<x>:ECOUNT:POLarity?  
<x> = 1 to 8

**Example** :MATH1:ECOUNT:POLARITY RISE  
:MATH1:ECOUNT:POLARITY? -> :MATH1:  
ECOUNT:POLARITY RISE

### **:MATH<x>:FILTer?**

**Function** Queries all settings related to the filter.

**Syntax** :MATH<x>:FILTer?  
<x> = 1 to 8

**Example** :MATH1:FILTER? -> :MATH1:FILTER:DELAY:  
TIME 0.000E+00;:MATH1:FILTER:IIR:  
FORDER 2;HIPASS:COFF 10.00E+06;:MATH1:  
FILTER:IIR:LOWPASS:COFF 10.00E+06;:  
MATH1:FILTER:MAVG:WEIGHT 2;:MATH1:  
FILTER:RESCALING:AVALUE 1.000E+00;  
BVALUE 0.000E+00;:MATH1:FILTER:  
TYPE THROUGH

### **:MATH<x>:FILTer:DELAy?**

**Function** Queries all settings related to the delay computation.

**Syntax** :MATH<x>:FILTer:DELAy?  
<x> = 1 to 8

**Example** :MATH1:FILTER:DELAY? -> :MATH1:FILTER:  
DELAY:TIME 0.000E+00

### **:MATH<x>:FILTer:DELAy:TIME**

**Function** Sets the delay value of the delay computation or queries the current setting.

**Syntax** :MATH<x>:FILTer:DELAy:TIME {<Time>}  
:MATH<x>:FILTer:DELAy:TIME?  
<x> = 1 to 8  
<Time> = See the DL9000 User's Manual.

**Example** :MATH1:FILTER:DELAY:TIME 1S  
:MATH1:FILTER:DELAY:TIME? -> :MATH1:  
FILTER:DELAY:TIME 1.000E+00

**:MATH<x>:FILTer:IIR?**

Function Queries all settings related to the IIR filter computation.

Syntax :MATH<x>:FILTer:IIR?  
<x> = 1 to 8

Example :MATH1:FILTer:IIR? -> :MATH1:FILTer:  
IIR:FORDER 2;HIPASS:COFF 10.00E+06;;  
MATH1:FILTer:IIR:LOWPASS:COFF 10.00E+00

**:MATH<x>:FILTer:IIR:FORDER (Filter Order)**

Function Sets the filter order of the IIR filter computation or queries the current setting.

Syntax :MATH<x>:FILTer:IIR:FORDER {<Nrf>}  
:MATH<x>:FILTer:IIR:FORDER?  
<x> = 1 to 8  
<Nrf> = 1 or 2

Example :MATH1:FILTer:IIR:FORDER 2  
:MATH1:FILTer:IIR:FORDER? -> :MATH1:  
FILTer:IIR:FORDER 2

**:MATH<x>:FILTer:IIR:HIPass?**

Function Queries all settings related to the IIR high pass filter computation.

Syntax :MATH<x>:FILTer:IIR:HIPass?  
<x> = 1 to 8

Example :MATH1:FILTer:IIR:HIPASS? -> :MATH1:  
FILTer:IIR:HIPASS:COFF 10.00E+06

**:MATH<x>:FILTer:IIR:HIPass:COFF**

Function Sets the cutoff frequency of the IIR high pass filter computation or queries the current setting.

Syntax :MATH<x>:FILTer:IIR:HIPass:  
COFF {<Frequency>}  
:MATH<x>:FILTer:IIR:HIPass:COFF?  
<x> = 1 to 8  
<Frequency> = 0.01 to 1 G (Hz)

Example :MATH1:FILTer:IIR:HIPASS:COFF 10MHZ  
:MATH1:FILTer:IIR:HIPASS:COFF?  
-> :MATH1:FILTer:IIR:HIPASS:  
COFF 10.00E+06

**:MATH<x>:FILTer:IIR:LOWPass?**

Function Queries all settings related to the IIR low pass filter computation.

Syntax :MATH<x>:FILTer:IIR:LOWPass?  
<x> = 1 to 8

Example :MATH1:FILTer:IIR:LOWPASS? -> :MATH1:  
FILTer:IIR:LOWPASS:  
COFF 10.00E+06

**:MATH<x>:FILTer:IIR:LOWPass:COFF**

Function Sets the cutoff frequency of the IIR low pass filter computation or queries the current setting.

Syntax :MATH<x>:FILTer:IIR:LOWPass:C  
OFF {<Frequency>}  
:MATH<x>:FILTer:IIR:LOWPass:COFF?  
<x> = 1 to 8  
<Frequency> = 0.01 to 1 G (Hz)

Example :MATH1:FILTer:IIR:LOWPASS:COFF 10HZ  
:MATH1:FILTer:IIR:LOWPASS:COFF?  
-> :MATH1:FILTer:IIR:LOWPASS:  
COFF 10.00E+00

**:MATH<x>:FILTer:MAVG?**

Function Queries all settings related to the moving average computation.

Syntax :MATH<x>:FILTer:MAVG?  
<x> = 1 to 8

Example :MATH1:FILTer:MAVG? -> :MATH1:FILTer:  
MAVG:WEIGHT 2

**:MATH<x>:FILTer:MAVG:WEIGHT**

Function Sets the weight of the moving average computation or queries the current setting.

Syntax :MATH<x>:FILTer:MAVG:WEIGHT {<Nrf>}  
:MATH<x>:FILTer:MAVG:WEIGHT?  
<x> = 1 to 8  
<Nrf> = 2 to 128 in 2<sup>n</sup> steps

Example :MATH1:FILTer:MAVG:WEIGHT 2  
:MATH1:FILTer:MAVG:WEIGHT? -> :MATH1:  
FILTer:MAVG:WEIGHT 2

**:MATH<x>:FILTer:RESCaling?**

Function Queries all settings related to the rescaling of the filter.

Syntax :MATH<x>:FILTer:RESCaling?  
<x> = 1 to 8

Example :MATH1:FILTer:RESCALING? -> :MATH1:  
FILTer:RESCALING:  
AVALUE 1.000E+00;BVALUE 0.000E+00

**:MATH<x>:FILTer:RESCaling:AVALue**

Function Sets rescaling coefficient A of the filter or queries the current setting.

Syntax :MATH<x>:FILTer:RESCaling:  
AVALue {<Nrf>}  
:MATH<x>:FILTer:RESCaling:AVALue?  
<x> = 1 to 8  
<Nrf> = -1.0000E+31 to 1.0000E+31

Example :MATH1:FILTer:RESCALING:AVALue 1  
:MATH1:FILTer:RESCALING:AVALue?  
-> :MATH1:FILTer:RESCALING:  
AVALue 1.000E+00

## 5.17 MATH Group

**:MATH<x>:FILTER:RESCaling:BVALue**  
Function Sets rescaling offset B of the filter or queries the current setting.  
Syntax :MATH<x>:FILTER:RESCaling:  
BVALue {<NRf>}  
:MATH<x>:FILTER:RESCaling:BVALue?  
<x> = 1 to 8  
<NRf> = -1.0000E+31 to 1.0000E+31  
Example :MATH1:FILTER:RESCALING:BVALUE 0  
:MATH1:FILTER:RESCALING:BVALUE?  
-> :MATH1:FILTER:RESCALING:  
BVALUE 0.000E+00

**:MATH<x>:FILTER:TYPE**  
Function Sets the filter type or queries the current setting.  
Syntax :MATH<x>:FILTER:TYPE {DElay|IHPass|  
ILPass|MAVG|THROUGH}  
:MATH<x>:FILTER:TYPE?  
<x> = 1 to 8  
Example :MATH1:FILTER:TYPE DELAY  
:MATH1:FILTER:TYPE? -> :MATH1:FILTER:  
TYPE DELAY

**:MATH<x>:INTEgral?**  
Function Queries all settings related to the integral computation.  
Syntax :MATH<x>:INTEgral?  
<x> = 1 to 8  
Example :MATH1:INTEGRAL? -> :MATH1:INTEGRAL:  
PSCALING:AVALUE 1.000E+00;  
BVALUE 0.000E+00;:MATH1:INTEGRAL:  
RESCALING:AVALUE 1.000E+00;  
BVALUE 0.000E+00

**:MATH<x>:INTEgral:PSCaling?**  
Function Queries all settings related to the pre-scaling of the integral computation.  
Syntax :MATH<x>:INTEgral:PSCaling?  
<x> = 1 to 8  
Example :MATH1:INTEGRAL:PSCALING? -> :MATH1:  
INTEGRAL:PSCALING:  
AVALUE 1.000E+00;BVALUE 0.000E+00

**:MATH<x>:INTEgral:PSCaling:AVALue**  
Function Sets pre-scaling coefficient A of the integral computation or queries the current setting.  
Syntax :MATH<x>:INTEgral:PSCaling:  
AVALue {<NRf>}  
:MATH<x>:INTEgral:PSCaling:AVALue?  
<x> = 1 to 8  
<NRf> = -1.0000E+31 to 1.0000E+31  
Example :MATH1:INTEGRAL:PSCALING:AVALUE 1  
:MATH1:INTEGRAL:PSCALING:AVALUE?  
-> :MATH1:INTEGRAL:PSCALING:  
AVALUE 1.000E+00

**:MATH<x>:INTEgral:PSCaling:BVALue**  
Function Sets pre-scaling offset B of the integral computation or queries the current setting.  
Syntax :MATH<x>:INTEgral:PSCaling:  
BVALue {<NRf>}  
:MATH<x>:INTEgral:PSCaling:BVALue?  
<x> = 1 to 8  
<NRf> = -1.0000E+31 to 1.0000E+31  
Example :MATH1:INTEGRAL:PSCALING:BVALUE 0  
:MATH1:INTEGRAL:PSCALING:BVALUE?  
-> :MATH1:INTEGRAL:PSCALING:  
BVALUE 0.000E+00

**:MATH<x>:INTEgral:RESCaling?**  
Function Queries all settings related to the rescaling of the integral computation.  
Syntax :MATH<x>:INTEgral:RESCaling?  
<x> = 1 to 8  
Example :MATH1:INTEGRAL:RESCALING? -> :MATH1:  
INTEGRAL:RESCALING:  
AVALUE 1.000E+00;BVALUE 0.000E+00

**:MATH<x>:INTEgral:RESCaling:AVALue**  
Function Sets rescaling coefficient A of the integral computation or queries the current setting.  
Syntax :MATH<x>:INTEgral:RESCaling:  
AVALue {<NRf>}  
:MATH<x>:INTEgral:RESCaling:AVALue?  
<x> = 1 to 8  
<NRf> = -1.0000E+31 to 1.0000E+31  
Example :MATH1:INTEGRAL:RESCALING:AVALUE 1  
:MATH1:INTEGRAL:RESCALING:AVALUE?  
-> :MATH1:INTEGRAL:RESCALING:  
AVALUE 1.000E+00

**:MATH<x>:INTEgral:RESCaling:BVALue**  
Function Sets rescaling offset B of the integral computation or queries the current setting.  
Syntax :MATH<x>:INTEgral:RESCaling:  
BVALue {<NRf>}  
:MATH<x>:INTEgral:RESCaling:BVALue?  
<x> = 1 to 8  
<NRf> = -1.0000E+31 to 1.0000E+31  
Example :MATH1:INTEGRAL:RESCALING:BVALUE 0  
:MATH1:INTEGRAL:RESCALING:BVALUE?  
-> :MATH1:INTEGRAL:RESCALING:  
BVALUE 0.000E+00

**:MATH<x>:INVErt**  
Function Turns ON/OFF the inverted display of the computed waveform or queries the current setting.  
Syntax :MATH<x>:INVErt {<Boolean>}  
:MATH<x>:INVErt?  
<x> = 1 to 8  
Example :MATH1:INVERT ON  
:MATH1:INVERT? -> :MATH1:INVERT 1

**:MATH<x>:IPOINT? (Initial Point)**

Function Queries all settings related to the computation reference point.

Syntax :MATH<x>:IPOINT?  
<x> = 1 to 8

Example :MATH1:IPOINT? -> :MATH1:IPOINT:  
POSITION 0.000E+00

**:MATH<x>:IPOINT:JUMP**

Function Moves the computation reference point to the specified position.

Syntax :MATH<x>:IPOINT:JUMP {<NRf>|TRIGger|Z1|Z2}  
<x> = 1 to 8  
<NRf> = -5 or 0 (div)

Example :MATH1:IPOINT:JUMP TRIGGER

**:MATH<x>:IPOINT:POSITION**

Function Sets the computation reference point or queries the current setting.

Syntax :MATH<x>:IPOINT:POSITION {<NRf>}  
:MATH<x>:IPOINT:POSITION?  
<x> = 1 to 8  
<NRf> = -5 to 5 (div)

Example :MATH1:IPOINT:POSITION 0  
:MATH1:IPOINT:POSITION? -> :MATH1:  
IPOINT:POSITION 0.000E+00

Description This command is valid when :MATH<x>:OPERation INTEGRal.

**:MATH<x>:LABEL?**

Function Queries all settings related to the label of the computed waveform.

Syntax :MATH<x>:LABEL?  
<x> = 1 to 8

Example :MATH1:LABEL? -> :MATH1:LABEL:  
DEFINE "MATH1";MODE 0

**:MATH<x>:LABEL[:DEFine]**

Function Sets the label of the computed waveform or queries the current setting.

Syntax :MATH<x>:LABEL[:DEFine] {<String>}  
:MATH<x>:LABEL[:DEFine]?  
<x> = 1 to 8  
<String> = Up to 8 characters

Example :MATH1:LABEL:DEFINE "MATH1"  
:MATH1:LABEL:DEFINE? -> :MATH1:LABEL:  
DEFINE "MATH1"

**:MATH<x>:LABEL:MODE**

Function Turns ON/OFF the label display of the computed waveform or queries the current setting.

Syntax :MATH<x>:LABEL:MODE {<Boolean>}  
:MATH<x>:LABEL:MODE?  
<x> = 1 to 8

**:MATH<x>:MINus?**

Function Queries all settings related to the subtraction.

Syntax :MATH<x>:MINus?  
<x> = 1 to 8

Example :MATH1:MINUS? -> :MATH1:MINUS:  
PSCALING1:AVALUE 1.000E+00;  
BVALUE 0.000E+00;:MATH1:MINUS:  
PSCALING2:AVALUE 1.000E+00;  
BVALUE 0.000E+00;:MATH1:MINUS:  
RESCALING:AVALUE 1.000E+00;  
BVALUE 0.000E+00+00

**:MATH<x>:MINus:PSCaling<x>?**

Function Queries all settings related to the pre-scaling of the subtraction.

Syntax :MATH<x>:MINus:PSCaling<x>?  
<x> of MATH<x> = 1 to 8  
<x> of PSCaling<x> = 1 or 2

Example :MATH1:MINUS:PSCALING1? -> :MATH1:  
MINUS:PSCALING1:  
AVALUE 1.000E+00;BVALUE 0.000E+00

**:MATH<x>:MINus:PSCaling<x>:AVALue**

Function Sets pre-scaling coefficient A of the subtraction or queries the current setting.

Syntax :MATH<x>:MINus:PSCaling<x>:  
AVALue {<NRf>}  
:MATH<x>:MINus:PSCaling<x>:AVALue?  
<x> of MATH<x> = 1 to 8  
<x> of PSCaling<x> = 1 or 2  
<NRf> = -1.0000E+31 to 1.0000E+31

Example :MATH1:MINUS:PSCALING1:AVALUE 1  
:MATH1:MINUS:PSCALING1:AVALUE?  
-> :MATH1:MINUS:PSCALING1:  
AVALUE 1.000E+00

**:MATH<x>:MINus:PSCaling<x>:BVALue**

Function Sets pre-scaling offset B of the subtraction or queries the current setting.

Syntax :MATH<x>:MINus:PSCaling<x>:  
BVALue {<NRf>}  
:MATH<x>:MINus:PSCaling<x>:BVALue?  
<x> of MATH<x> = 1 to 8  
<x> of PSCaling<x> = 1 or 2  
<NRf> = -1.0000E+31 to 1.0000E+31

Example :MATH1:MINUS:PSCALING1:BVALUE 0  
:MATH1:MINUS:PSCALING1:BVALUE?  
-> :MATH1:MINUS:PSCALING1:  
BVALUE 0.000E+00

## 5.17 MATH Group

### **:MATH<x>:MINus:RESCaling?**

**Function** Queries all settings related to the rescaling of the subtraction.

**Syntax** :MATH<x>:MINus:RESCaling?  
<x> = 1 to 8

**Example** :MATH1:MINus:RESCALING? -> :MATH1:MINus:  
RESCALING:  
AVALUE 1.000E+00;BVALUE 0.000E+00

### **:MATH<x>:MINus:RESCaling:AVALue**

**Function** Sets rescaling coefficient A of the subtraction or queries the current setting.

**Syntax** :MATH<x>:MINus:RESCaling:AVALue {<NRf>}  
:MATH<x>:MINus:RESCaling:AVALue?  
<x> = 1 to 8  
<NRf> = -1.0000E+31 to 1.0000E+31

**Example** :MATH1:MINus:RESCALING:AVALUE 1  
:MATH1:MINus:RESCALING:AVALUE? -> :MATH1:  
MINus:RESCALING:AVALUE 1.000E+00

### **:MATH<x>:MINus:RESCaling:BVALue**

**Function** Sets rescaling offset B of the subtraction or queries the current setting.

**Syntax** :MATH<x>:MINus:RESCaling:BVALue {<NRf>}  
:MATH<x>:MINus:RESCaling:BVALue?  
<x> = 1 to 8  
<NRf> = -1.0000E+31 to 1.0000E+31

**Example** :MATH1:MINus:RESCALING:BVALUE 0  
:MATH1:MINus:RESCALING:BVALUE? -> :MATH1:  
MINus:RESCALING:BVALUE 0.000E+00

### **:MATH<x>:MULTiple?**

**Function** Queries all settings related to the multiplication.

**Syntax** :MATH<x>:MULTiple?  
<x> = 1 to 8

**Example** :MATH1:MULTIPLE? -> :MATH1:MULTIPLE:  
PSCALING1:AVALUE 1.000E+00;  
BVALUE 0.000E+00;:MATH1:MULTIPLE:  
PSCALING2:AVALUE 1.000E+00;  
BVALUE 0.000E+00;:MATH1:MULTIPLE:  
RESCALING:AVALUE 1.000E+00;  
BVALUE 0.000E+00

### **:MATH<x>:MULTiple:PSCaling<x>?**

**Function** Queries all settings related to the pre-scaling of the multiplication.

**Syntax** :MATH<x>:MULTiple:PSCaling<x>?  
<x> of MATH<x> = 1 to 8  
<x> of PSCaling<x> = 1 or 2

**Example** :MATH1:MULTIPLE:PSCALING1? -> :MATH1:  
MULTIPLE:PSCALING1:  
AVALUE 1.000E+00;BVALUE 0.000E+00

### **:MATH<x>:MULTiple:PSCaling<x>:AVALue**

**Function** Sets pre-scaling coefficient A of the multiplication or queries the current setting.

**Syntax** :MATH<x>:MULTiple:PSCaling<x>:  
AVALue {<NRf>}  
:MATH<x>:MULTiple:PSCaling<x>:AVALue?  
<x> of MATH<x> = 1 to 8  
<x> of PSCaling<x> = 1 or 2  
<NRf> = -1.0000E+31 to 1.0000E+31

**Example** :MATH1:MULTIPLE:PSCALING1:AVALUE 1  
:MATH1:MULTIPLE:PSCALING1:AVALUE?  
-> :MATH1:MULTIPLE:PSCALING1:  
AVALUE 1.000E+00

### **:MATH<x>:MULTiple:PSCaling<x>:BVALue**

**Function** Sets pre-scaling offset B of the multiplication or queries the current setting.

**Syntax** :MATH<x>:MULTiple:PSCaling<x>:  
BVALue {<NRf>}  
:MATH<x>:MULTiple:PSCaling<x>:BVALue?  
<x> of MATH<x> = 1 to 8  
<x> of PSCaling<x> = 1 or 2  
<NRf> = -1.0000E+31 to 1.0000E+31

**Example** :MATH1:MULTIPLE:PSCALING1:BVALUE 0  
:MATH1:MULTIPLE:PSCALING1:BVALUE?  
-> :MATH1:MULTIPLE:PSCALING1:  
BVALUE 0.000E+00

### **:MATH<x>:MULTiple:RESCaling?**

**Function** Queries all settings related to the rescaling of the multiplication.

**Syntax** :MATH<x>:MULTiple:RESCaling?  
<x> = 1 to 8

**Example** :MATH1:MULTIPLE:RESCALING? -> :MATH1:  
MULTIPLE:RESCALING:  
AVALUE 1.000E+00;BVALUE 0.000E+00

### **:MATH<x>:MULTiple:RESCaling:AVALue**

**Function** Sets rescaling coefficient A of the multiplication or queries the current setting.

**Syntax** :MATH<x>:MULTiple:RESCaling:  
AVALue {<NRf>}  
:MATH<x>:MULTiple:RESCaling:AVALue?  
<x> = 1 to 8  
<NRf> = -1.0000E+31 to 1.0000E+31

**Example** :MATH1:MULTIPLE:RESCALING:AVALUE 1  
:MATH1:MULTIPLE:RESCALING:AVALUE?  
-> :MATH1:MULTIPLE:RESCALING:  
AVALUE 1.000E+00



**:MATH<x>:MULTiple:RESCaling:BVALue**

**Function** Sets rescaling offset B of the multiplication or queries the current setting.

**Syntax** :MATH<x>:MULTiple:RESCaling:  
BVALue {<Nrf>}  
:MATH<x>:MULTiple:RESCaling:BVALue?  
<x> = 1 to 8  
<Nrf> = -1.0000E+31 to 1.0000E+31

**Example** :MATH1:MULTIPLE:RESCALING:BVALUE 0  
:MATH1:MULTIPLE:RESCALING:BVALUE?  
-> :MATH1:MULTIPLE:RESCALING:  
BVALUE 0.000E+00

**:MATH<x>:OPERation**

**Function** Sets the operator or queries the current setting.

**Syntax** :MATH<x>:OPERation {(ECount|FILTer|  
INTEgral|MINus|MULTiple|PLUS|RCOunt|  
USERdefine),<Nrf>,<Nrf>}  
:MATH<x>:OPERation?  
<x> = 1 to 8 (<x> = 1 to 4 only when {USERdefine}  
selected)  
<Nrf> = 1 to 4

**Example** :MATH1:OPERATION FILTER,1  
:MATH1:OPERATION? ->  
:MATH1:OPERATION FILTER,1

**Description** For unary operators (ECount|FILTer|INTEgral), select the target waveform using the first <Nrf>.

For binary operators (MINus|MULTiple|PLUS|RCOunt), select the target waveform of the first term using the first <Nrf> and the target waveform of the second term using the second <Nrf>.

<Nrf> is not required for the USERdefine operator.

**:MATH<x>:PLUS?**

**Function** Queries all settings related to the addition.

**Syntax** :MATH<x>:PLUS?  
<x> = 1 to 8

**Example** :MATH1:PLUS? -> :MATH1:PLUS:  
PSCALING1:AVALUE 1.000E+00;  
BVALUE 0.000E+00;:MATH1:PLUS:PSCALING2:  
AVALUE 1.000E+00;BVALUE 0.000E+00;:  
MATH1:PLUS:RESCALING:AVALUE 1.000E+00;  
BVALUE 0.000E+00

**:MATH<x>:PLUS:PSCaling<x>?**

**Function** Queries all settings related to the pre-scaling of the addition.

**Syntax** :MATH<x>:PLUS:PSCaling<x>?  
<x> of MATH<x> = 1 to 8  
<x> of PSCaling<x> = 1 or 2

**Example** :MATH1:PLUS:PSCALING1? -> :MATH1:PLUS:  
PSCALING1:  
AVALUE 1.000E+00;BVALUE 0.000E+00

**:MATH<x>:PLUS:PSCaling<x>:AVALue**

**Function** Sets pre-scaling coefficient A of the addition or queries the current setting.

**Syntax** :MATH<x>:PLUS:PSCaling<x>:  
AVALue {<Nrf>}  
:MATH<x>:PLUS:PSCaling<x>:AVALue?  
<x> of MATH<x> = 1 to 8  
<x> of PSCaling<x> = 1 or 2  
<Nrf> = -1.0000E+31 to 1.0000E+31

**Example** :MATH1:PLUS:PSCALING1:AVALUE 1  
:MATH1:PLUS:PSCALING1:AVALUE?  
-> :MATH1:PLUS:PSCALING1:  
AVALUE 1.000E+00

**:MATH<x>:PLUS:PSCaling<x>:BVALue**

**Function** Sets pre-scaling offset B of the addition or queries the current setting.

**Syntax** :MATH<x>:PLUS:PSCaling<x>:  
BVALue {<Nrf>}  
:MATH<x>:PLUS:PSCaling<x>:BVALue?  
<x> of MATH<x> = 1 to 8  
<x> of PSCaling<x> = 1 or 2  
<Nrf> = -1.0000E+31 to 1.0000E+31

**Example** :MATH1:PLUS:PSCALING1:BVALUE 0  
:MATH1:PLUS:PSCALING1:BVALUE?  
-> :MATH1:PLUS:PSCALING1:BVALUE 0.000E+00

**:MATH<x>:PLUS:RESCaling?**

**Function** Queries all settings related to the rescaling of the addition.

**Syntax** :MATH<x>:PLUS:RESCaling?  
<x> = 1 to 8

**Example** :MATH1:PLUS:RESCALING? -> :MATH1:PLUS:  
RESCALING:AVALUE 1.000E+00;  
BVALUE 0.000E+00

**:MATH<x>:PLUS:RESCaling:AVALue**

**Function** Sets rescaling coefficient A of the addition or queries the current setting.

**Syntax** :MATH<x>:PLUS:RESCaling:AVALue {<Nrf>}  
:MATH<x>:PLUS:RESCaling:AVALue?  
<x> = 1 to 8  
<Nrf> = -1.0000E+31 to 1.0000E+31

**Example** :MATH1:PLUS:RESCALING:AVALUE 1  
:MATH1:PLUS:RESCALING:AVALUE?  
-> :MATH1:PLUS:RESCALING:AVALUE 1.000E+00

**:MATH<x>:PLUS:RESCaling:BVALue**

**Function** Sets rescaling offset B of the addition or queries the current setting.

**Syntax** :MATH<x>:PLUS:RESCaling:BVALue {<Nrf>}  
:MATH<x>:PLUS:RESCaling:BVALue?  
<x> = 1 to 8  
<Nrf> = -1.0000E+31 to 1.0000E+31

**Example** :MATH1:PLUS:RESCALING:BVALUE 0  
:MATH1:PLUS:RESCALING:BVALUE?  
-> :MATH1:PLUS:RESCALING:  
BVALUE 0.000E+00

## 5.17 MATH Group

### **:MATH<x>:POSITION**

**Function** Sets the vertical position of the computed waveform or queries the current setting.

**Syntax** :MATH<x>:POSITION {<NRf>}  
:MATH<x>:POSITION?  
<x> = 1 to 8  
<NRf> = -4 to 4 (div)

**Example** :MATH1:POSITION 0  
:MATH1:POSITION? -> :MATH1:  
POSITION 0.000E+00

### **:MATH<x>:SCALE?**

**Function** Queries all settings related to scaling.

**Syntax** :MATH<x>:SCALE?  
<x> = 1 to 8

**Example** :MATH1:SCALE? -> :MATH1:SCALE:  
CENTER 1.000E+00;MODE AUTO;  
SENSITIVITY 1.000E+00

### **:MATH<x>:SCALE:CENTER**

**Function** Sets the offset of the computed waveform or queries the current setting.

**Syntax** :MATH<x>:SCALE:CENTER {<NRf>|<Voltage>|  
<Current>}  
:MATH<x>:SCALE:CENTER?  
<x> = 1 to 8  
<NRf>, <Voltage>, and <Current> = See the DL9000  
User's Manual.

**Example** :MATH1:SCALE:CENTER 1  
:MATH1:SCALE:CENTER? -> :MATH1:SCALE:  
CENTER 1.000E+00

### **:MATH<x>:SCALE:MODE**

**Function** Sets the scaling mode or queries the current setting.

**Syntax** :MATH<x>:SCALE:MODE {AUTO|MANual}  
:MATH<x>:SCALE:MODE?  
<x> = 1 to 8

**Example** :MATH1:SCALE:MODE AUTO  
:MATH1:SCALE:MODE? -> :MATH1:SCALE:  
MODE AUTO

### **:MATH<x>:SCALE:SENSitivity**

**Function** Sets the vertical sensitivity of the computed waveform or queries the current setting.

**Syntax** :MATH<x>:SCALE:SENSitivity {<NRf>|  
<Voltage>|<Current>}  
:MATH<x>:SCALE:SENSitivity?  
<x> = 1 to 8  
<NRf>, <Voltage>, and <Current> = See the DL9000  
User's Manual.

**Example** :MATH1:SCALE:SENSITIVITY 1  
:MATH1:SCALE:SENSITIVITY? -> :MATH1:  
SCALE:SENSITIVITY 1.000E+00

### **:MATH<x>:SELEct**

**Function** Sets the display option or queries the current setting.

**Syntax** :MATH<x>:SELEct {INPut|MATH|REFerence}  
:MATH<x>:SELEct?  
<x> = 1 to 8

**Example** :MATH1:SELECT INPUT  
:MATH1:SELECT? -> :MATH1:SELECT INPUT

### **:MATH<x>:SVALue (Scale VALUE)**

**Function** Turns ON/OFF the scale value display or queries the current setting.

**Syntax** :MATH<x>:SVALue {<Boolean>}  
:MATH<x>:SVALue?  
<x> = 1 to 8

**Example** :MATH1:SVALUE ON  
:MATH1:SVALUE? -> MATH1:SVALUE 1

### **:MATH<x>:THReshold<x>**

**Function** Sets the edge detection level of the count computation or queries the current setting.

**Syntax** :MATH<x>:THReshold<x> {<NRf>|  
<Voltage>|<Current>}  
:MATH<x>:THReshold<x>?  
<x> of MATH<x> = 1 to 8  
<x> of THReshold<x> = 1 or 2  
<NRf>, <Voltage>, and <Current> = See the DL9000  
User's Manual.

**Example** :MATH1:THRESHOLD1 1  
:MATH1:THRESHOLD1? -> :MATH1:  
THRESHOLD1 1.000E+00

Description THReshold2 is valid when the operation is {RCOUNT}.

### **:MATH<x>:UNIT?**

**Function** Queries all settings related to the computation unit.

**Syntax** :MATH<x>:UNIT?  
<x> = 1 to 8

**Example** :MATH1:UNIT? ->  
:MATH1:UNIT:DEFINE "EU";MODE AUTO

### **:MATH<x>:UNIT[:DEFine]**

**Function** Sets the computation unit or queries the current setting.

**Syntax** :MATH<x>:UNIT[:DEFine] {<String>}  
:MATH<x>:UNIT[:DEFine]?  
<x> = 1 to 8  
<String> = Up to 4 characters

**Example** :MATH1:UNIT:DEFINE "EU"  
:MATH1:UNIT:DEFINE? -> :MATH1:UNIT:  
DEFINE "EU"

**:MATH<x>:UNIT:MODE**

Function Sets the automatic/manual addition of the computation unit or queries the current setting.

Syntax :MATH<x>:UNIT:MODE {AUTO|USERdefine}  
:MATH<x>:UNIT:MODE?  
<x> = 1 to 8

Example :MATH1:UNIT:MODE AUTO  
:MATH1:UNIT:MODE? -> :MATH1:UNIT:  
MODE AUTO

**:MATH<x>:USERdefine?**

Function Queries all settings related to user-defined math or queries the current setting.

Syntax :MATH<x>:USERdefine?  
<x> = 1 to 4

Example :MATH1:USERDEFINE? -> :MATH1:  
USERDEFINE:CONSTANT1 1.000E+00;  
CONSTANT2 1.000E+00;  
CONSTANT3 1.000E+00;  
CONSTANT4 1.000E+00;DEFINE "C1-C2"

**:MATH<x>:USERdefine:ARANGing**

Function Executes auto ranging for user-defined math.

Syntax :MATH<x>:USERdefine:ARANGing  
<x> = 1 to 4

Example :MATH1:USERDEFINE:ARANGING

**:MATH<x>:USERdefine:CONSTant<x>**

Function Sets a user-defined math constant or queries the current setting.

Syntax :MATH<x>:USERdefine:CONSTant<x> {<Nrf>}  
:MATH<x>:USERdefine:CONSTant<x>?  
MATH<x>: <x> = 1 to 4  
CONSTant<x>: <x> = 1 to 4  
<Nrf> = -1.0000E+31 to 1.0000E+31

Example :MATH1:USERDEFINE:CONSTANT1 1  
:MATH1:USERDEFINE:CONSTANT1? -> :MATH1:  
USERDEFINE:CONSTANT1 1.000E+00

**:MATH<x>:USERdefine:DEFine**

Function Sets a user-defined math equation or queries the current setting.

Syntax :MATH<x>:USERdefine:DEFine {<string>}  
:MATH<x>:USERdefine:DEFine?  
<x> = 1 to 4  
<string> = 128 characters or less

Example :MATH1:USERDEFINE:DEFINE "C1-C2"  
:MATH1:USERDEFINE:DEFINE? -> :MATH1:  
USERDEFINE:DEFINE "C1-C2"

Description Characters and symbols other than those on the keyboard displayed on the main unit screen cannot be used.

See the main unit user's manual for details about equations.

**:MATH<x>:USERdefine:HISTORY:ABORT**

Function Cancels history computation for user-defined math.

Syntax :MATH<x>:USERdefine:HISTORY:ABORT  
<x> = 1 to 4

Example :MATH1:USERDEFINE:HISTORY:ABORT

**:MATH<x>:USERdefine:HISTORY:EXECute**

Function Executes history computation for user-defined math.

Syntax :MATH<x>:USERdefine:HISTORY:EXECute  
<x> = 1 to 4

Example :MATH1:USERDEFINE:HISTORY:EXECUTE

## 5.18 MEASure Group

### :MEASure?

**Function** Queries all settings related to the automated measurement of waveform parameters.

**Syntax** :MEASure?

**Example** :MEASURE? -> :MEASURE:CALCULATION:  
 DEFINE1 "MAX(C1)";  
 DEFINE2 "MIN(C2)";DEFINE3 "HIGH(C3)";  
 DEFINE4 "LOW(C4)";STATE1 0;STATE2 0;  
 STATE3 0;STATE4 0;:MEASURE:CONTINUOUS:  
 COUNT 0;:MEASURE:CYCLE:TRACE 1;:  
 MEASURE:DISPLAY 1;MODE BASIC;THRESHOLD:  
 TRACE1:AUTO PTOPEAK;LHYSTERESIS:  
 HYSTERESIS 1.000E+00;LEVEL 0.000E+00;:  
 MEASURE:THRESHOLD:TRACE1:MODE AUTO;  
 ULOWER:RANGE 2.000E+00,1.000E+00;:  
 MEASURE:THRESHOLD:TRACE2:AUTO PTOPEAK;  
 LHYSTERESIS:HYSTERESIS 1.000E+00;  
 LEVEL 0.000E+00;:MEASURE:THRESHOLD:  
 TRACE2:MODE AUTO;ULOWER:  
 RANGE 2.000E+00,1.000E+00;:MEASURE:  
 THRESHOLD:TRACE3:AUTO PTOPEAK;  
 LHYSTERESIS:HYSTERESIS 1.000E+00;  
 LEVEL 0.000E+00;:MEASURE:THRESHOLD:  
 TRACE3:MODE AUTO;ULOWER:  
 RANGE 2.000E+00,1.000E+00;:MEASURE:  
 THRESHOLD:TRACE4:AUTO PTOPEAK;  
 LHYSTERESIS:HYSTERESIS 1.000E+00;  
 LEVEL 0.000E+00;:MEASURE:THRESHOLD:  
 TRACE4:MODE AUTO;ULOWER:  
 RANGE 2.000E+00,1.000E+00;:MEASURE:  
 THRESHOLD:TRACE5:AUTO PTOPEAK;  
 LHYSTERESIS:HYSTERESIS 1.000E+00;  
 LEVEL 0.000E+00;:MEASURE:THRESHOLD:  
 TRACE5:MODE AUTO;ULOWER:RANGE .....

### :MEASure:CALCulation?

**Function** Queries all settings related to calculation items.

**Syntax** :MEASure:CALCulation?

**Example** :MEASURE:CALCULATION? -> :MEASURE:  
 CALCULATION:DEFINE1 "MAX(C1)";  
 DEFINE2 "MIN(C2)";DEFINE3 "HIGH(C3)";  
 DEFINE4 "LOW(C4)";STATE1 0;STATE2 0;  
 STATE3 0;STATE4 0

### :MEASure:CALCulation:ALL

**Function** Turns ON/OFF all calculation items.

**Syntax** :MEASure:CALCulation:ALL {<Boolean>}

**Example** :MEASURE:CALCULATION:ALL ON

### :MEASure:CALCulation:COUNT<x>?

**Function** Queries the statistical processing count of the calculation item.

**Syntax** :MEASure:CALCulation:COUNT<x>?  
 <x> = 1 to 4

**Example** :MEASURE:CALCULATION:COUNT1?  
 -> :MEASURE:CALCULATION:COUNT1 1

### :MEASure:CALCulation:DEFine<x>

**Function** Sets the equation of the calculation item or queries the current setting.

**Syntax** :MEASure:CALCulation:DEFine<x>  
 {<String>}  
 :MEASure:CALCulation:DEFine<x>?  
 <x> = 1 to 4  
 <String> = Up to 128 characters

**Example** :MEASURE:CALCULATION:DEFINE1 "MAX(C1)"  
 :MEASURE:CALCULATION:DEFINE1? -> :  
 MEASURE:CALCULATION:DEFINE1 "MAX(C1)"

### :MEASure:CALCulation:{MAXimum<x>|MEAN<x>|MINimum<x>|SDEViation<x>}?

**Function** Queries the statistical value of the calculation item.

**Syntax** :MEASure:CALCulation:{MAXimum<x>|  
 MEAN<x>|MINimum<x>|SDEViation<x>}?  
 <x> of MAXimum<x> = 1 to 4  
 <x> of MEAN<x> = 1 to 4  
 <x> of MINimum<x> = 1 to 4  
 <x> of SDEViation<x> = 1 to 4

**Example** (The following is an example for the maximum value.)  
 :MEASURE:CALCULATION:MAXIMUM1?  
 -> :MEASURE:CALCULATION:  
 MAXIMUM1 1.000E+00

### :MEASure:CALCulation:STATE<x>

**Function** Turns ON/OFF the calculation item or queries the current setting.

**Syntax** :MEASure:CALCulation:STATE<x>  
 {<Boolean>}  
 :MEASure:CALCulation:STATE<x>?  
 <x> = 1 to 4

**Example** :MEASURE:CALCULATION:STATE1 ON  
 :MEASURE:CALCULATION:STATE1?  
 -> :MEASURE:CALCULATION:STATE1 1

**:MEASure:CALCulation:VALue<x>?**

Function Queries the automated measured value of the calculation item.

Syntax :MEASure:CALCulation:VALue<x>? {<Nrf>}  
<x> = 1 to 4  
<Nrf> = 1 to 100000

Example :MEASURE:CALCULATION:VALUE1?  
-> :MEASURE:CALCULATION:VALUE1 1.000E+00

Description

- If the measurement is not possible, "NAN (Not A Number)" is returned.
- <Nrf> indicates the nth automated measured value in the past.  
In the case of cycle statistical processing, specify the <Nrf><sup>th</sup> cycle from the left of the screen.  
To specify the oldest automated measured value, specify 1.  
If <Nrf> is omitted, the latest automated measured value is specified.  
If the value corresponding to the relevant count is not present, "NAN (Not A Number)" is returned.

**:MEASure:CONTInuous?**

Function Queries all settings related to the continuous statistical processing.

Syntax :MEASure:CONTInuous?  
Example :MEASURE:CONTINUOUS? -> :MEASURE:CONTINUOUS:COUNT 0

**:MEASure:CONTInuous:COUNT**

Function Sets the continuous statistical processing count or queries the current setting.

Syntax :MEASure:CONTInuous:COUNT {<Nrf>}  
:MEASURE:CONTINUOUS:COUNT?  
<Nrf> = 0 to 100000

Example :MEASURE:CONTINUOUS:COUNT 10  
:MEASURE:CONTINUOUS:COUNT? -> :MEASURE:CONTINUOUS:COUNT 10

Description When <Nrf> = 0, the maximum count that is possible under the current settings is automatically set.

**:MEASure:CONTInuous:REStArt**

Function Restarts the continuous statistical processing.

Syntax :MEASure:CONTInuous:REStArt  
Example :MEASURE:CONTINUOUS:RESTART

Description Clears the previous statistical data.

**:MEASure:CYCLE?**

Function Queries all settings related to the cycle statistical processing.

Syntax :MEASure:CYCLE?  
Example :MEASURE:CYCLE? -> :MEASURE:CYCLE:TRACE 1

**:MEASure:CYCLE:ABORT**

Function Aborts the execution of the cycle statistical processing.

Syntax :MEASure:CYCLE:ABORT  
Example :MEASURE:CYCLE:ABORT

**:MEASure:CYCLE:EXECute**

Function Executes the cycle statistical processing. This is an overlap command.

Syntax :MEASure:CYCLE:EXECute  
Example :MEASURE:CYCLE:EXECUTE

Description Continues the operation without clearing the previous statistical data.

**:MEASure:CYCLE:TRACe**

Function Sets the cycle source trace of the continuous statistical processing count or queries the current setting.

Syntax :MEASure:CYCLE:TRACe {<Nrf>}  
:MEASURE:CYCLE:TRACe?  
<Nrf> = 1 to 8

Example :MEASURE:CYCLE:TRACE 1  
:MEASURE:CYCLE:TRACE? -> :MEASURE:CYCLE:TRACE 1

**:MEASure:DISPlay**

Function Turns ON/OFF the display of the automated measurement of waveform parameters or queries the current setting.

Syntax :MEASure:DISPlay {<Boolean>}  
:MEASURE:DISPLAY?

Example :MEASURE:DISPLAY ON  
:MEASURE:DISPLAY? -> :MEASURE:DISPLAY 1

**:MEASure:HISTory:ABORT**

Function Aborts the execution of the statistical processing of the history data.

Syntax :MEASure:HISTory:ABORT  
Example :MEASURE:HISTORY:ABORT

**:MEASure:HISTory:EXECute**

Function Executes the statistical processing of the history data. This is an overlap command.

Syntax :MEASure:HISTory:EXECute  
Example :MEASURE:HISTORY:EXECUTE

**:MEASure:MODE**

Function Sets the mode of the automated measurement of waveform parameters or queries the current setting.

Syntax :MEASure:MODE {BASIC|CONTInuous|CYCLE|HISTory}  
:MEASure:MODE?

Example :MEASURE:MODE BASIC  
:MEASURE:MODE? -> :MEASURE:MODE BASIC

## 5.18 MEASure Group

### **:MEASure:THReshold?**

**Function** Queries all settings related to the threshold levels of the automated measurement of waveform parameters.

**Syntax** :MEASure:THReshold?

**Example** :MEASURE:THRESHOLD? -> :MEASURE:  
THRESHOLD:TRACE1:AUTO PTOPEAK;  
LHYSTERESIS:HYSTERESIS 1.000E+00;  
LEVEL 0.000E+00;;MEASURE:THRESHOLD:  
TRACE1:MODE AUTO;ULOWER:  
RANGE 2.000E+00,1.000E+00;;MEASURE:  
THRESHOLD:TRACE2:AUTO PTOPEAK;  
LHYSTERESIS:HYSTERESIS 1.000E+00;  
LEVEL 0.000E+00;;MEASURE:THRESHOLD:  
TRACE2:MODE AUTO;ULOWER:  
RANGE 2.000E+00,1.000E+00;;MEASURE:  
THRESHOLD:TRACE3:AUTO PTOPEAK;  
LHYSTERESIS:HYSTERESIS 1.000E+00;  
LEVEL 0.000E+00;;MEASURE:THRESHOLD:  
TRACE3:MODE AUTO;ULOWER:  
RANGE 2.000E+00,1.000E+00;;MEASURE:  
THRESHOLD:TRACE4:AUTO PTOPEAK;  
LHYSTERESIS:HYSTERESIS 1.000E+00;  
LEVEL 0.000E+00;;MEASURE:THRESHOLD:  
TRACE4:MODE AUTO;ULOWER:  
RANGE 2.000E+00,1.000E+00;;MEASURE:  
THRESHOLD:TRACE5:AUTO PTOPEAK;  
LHYSTERESIS:HYSTERESIS 1.000E+00;  
LEVEL 0.000E+00;;MEASURE:THRESHOLD:  
TRACE5:MODE AUTO;ULOWER:  
RANGE 2.000E+00,1.000E+00;;MEASURE:  
THRESHOLD:TRACE6:AUTO PTOPEAK;  
LHYSTERESIS:HYSTERESIS 1.000E+00;  
LEVEL 0.000E+00;;MEASURE:THRESHOLD:  
TRACE6:MODE AUTO;ULOWER:RANGE .....

**Description** This command is valid when the <Parameter> of :MEASure:TRACe<x>:AREA<x>:<Parameter> is not RISE or FALL.

### **:MEASure:THReshold:TRACe<x>?**

**Function** Queries the threshold levels of the trace.

**Syntax** :MEASure:THReshold:TRACe<x>?

<x> = 1 to 8

**Example** :MEASURE:THRESHOLD:TRACE1? -> :MEASURE:  
THRESHOLD:TRACE1:AUTO PTOPEAK;  
LHYSTERESIS:HYSTERESIS 1.000E+00;  
LEVEL 0.000E+00;;MEASURE:THRESHOLD:  
TRACE1:MODE AUTO;ULOWER:  
RANGE 1.000E+00,2.000E+00

**Description** This command is valid when the <Parameter> of :MEASure:TRACe<x>:AREA<x>:<Parameter> is not RISE or FALL.

### **:MEASure:THReshold:TRACe<x>:AUTO**

**Function** Sets the detection mode when the auto setting of the threshold level is enabled or queries the current setting.

**Syntax** :MEASure:THReshold:TRACe<x>:AUTO {HLOW|PTOPeak}

:MEASure:THReshold:TRACe<x>:AUTO?

<x> = 1 to 8

**Example** :MEASURE:THRESHOLD:TRACE1:AUTO PTOPEAK  
:MEASURE:THRESHOLD:TRACE1:AUTO?  
-> :MEASURE:THRESHOLD:TRACE1:  
AUTO PTOPEAK

**Description** This command is valid when the <Parameter> of :MEASure:TRACe<x>:AREA<x>:<Parameter> is not RISE or FALL.

### **:MEASure:THReshold:TRACe<x>:**

#### **LHYSTeresis?**

**Function** Queries all settings related to the level and hysteresis of the threshold level.

**Syntax** :MEASure:THReshold:TRACe<x>:

LHYSTeresis?

<x> = 1 to 8

**Example** :MEASURE:THRESHOLD:TRACE1:LHYSTERESIS?  
-> :MEASURE:THRESHOLD:TRACE1:  
LHYSTERESIS:HYSTERESIS 1.000E+00;  
LEVEL 0.000E+00

**Description** This command is valid when the <Parameter> of :MEASure:TRACe<x>:AREA<x>:<Parameter> is not RISE or FALL.

### **:MEASure:THReshold:TRACe<x>:**

#### **LHYSTeresis:HYSTeresis**

**Function** Sets the hysteresis of the threshold level or queries the current setting.

**Syntax** :MEASure:THReshold:TRACe<x>:

LHYSTeresis:HYSTeresis {<Nrf>}

:MEASure:THReshold:TRACe<x>:

LHYSTeresis:HYSTeresis?

<x> = 1 to 8

<Nrf> = 0 to 4 (div)

**Example** :MEASURE:THRESHOLD:TRACE1:LHYSTERESIS:  
HYSTERESIS 1  
:MEASURE:THRESHOLD:TRACE1:LHYSTERESIS:  
HYSTERESIS? -> :MEASURE:THRESHOLD:  
TRACE1:  
LHYSTERESIS:HYSTERESIS 1.000E+00

**Description** This command is valid when the <Parameter> of :MEASure:TRACe<x>:AREA<x>:<Parameter> is not RISE or FALL.

**:MEASure:THReshold:TRACe<x>:****LHYSteresis:LEVel**

**Function** Sets the level of the threshold level or queries the current setting.

**Syntax** :MEASure:THReshold:TRACe<x>:  
LHYSteresis:LEVel {<NRf>|<Voltage>|<Current>}  
:MEASure:THReshold:TRACe<x>:  
LHYSteresis:LEVel?  
<x> = 1 to 8  
<NRf>, <Voltage>, and <Current> = See the DL9000 User's Manual.

**Example** :MEASURE:THRESHOLD:TRACE1:LHYSTERESIS:  
LEVEL 1  
:MEASURE:THRESHOLD:TRACE1:LHYSTERESIS:  
LEVEL? -> :MEASURE:THRESHOLD:TRACE1:  
LHYSTERESIS:LEVEL 1.000E+00

**Description** This command is valid when the <Parameter> of :MEASure:TRACe<x>:AREA<x>:<Parameter> is not RISE or FALL.

**:MEASure:THReshold:TRACe<x>:MODE**

**Function** Sets the setup mode of the threshold level or queries the current setting.

**Syntax** :MEASure:THReshold:TRACe<x>:MODE {AUTO|LHYSteresis|ULOWer}  
:MEASure:THReshold:TRACe<x>:MODE?  
<x> = 1 to 8

**Example** :MEASURE:THRESHOLD:TRACE1:  
MODE LHYSTERESIS  
:MEASURE:THRESHOLD:TRACE1:MODE?  
-> :MEASURE:THRESHOLD:TRACE1:  
MODE LHYSTERESIS

**Description** This command is valid when the <Parameter> of :MEASure:TRACe<x>:AREA<x>:<Parameter> is not RISE or FALL.

**:MEASure:THReshold:TRACe<x>:ULOWer?**

**Function** Queries all settings related to the upper and lower limits of the threshold level.

**Syntax** :MEASure:THReshold:TRACe<x>:ULOWer?  
<x> = 1 to 8

**Example** :MEASURE:THRESHOLD:TRACE1:ULOWER?  
-> :MEASURE:THRESHOLD:TRACE1:ULOWER:  
RANGE 2.000E+00,1.000E+00

**Description** This command is valid when the <Parameter> of :MEASure:TRACe<x>:AREA<x>:<Parameter> is not RISE or FALL.

**:MEASure:THReshold:TRACe<x>:ULOWer:****RANGE**

**Function** Sets the upper and lower limits of the threshold level or queries the current setting.

**Syntax** :MEASure:THReshold:TRACe<x>:ULOWer:  
RANGE {(<NRf>,<NRf>)|  
(<Voltage>,<Voltage>)|  
(<Current>,<Current>)}  
:MEASure:THReshold:TRACe<x>:ULOWer:  
RANGE?  
<x> = 1 to 8  
<NRf>, <Voltage>, and <Current> = See the DL9000 User's Manual.

**Example** :MEASURE:THRESHOLD:TRACE1:ULOWER:  
RANGE 1,2  
:MEASURE:THRESHOLD:TRACE1:ULOWER:RANGE?  
-> :MEASURE:THRESHOLD:TRACE1:ULOWER:  
RANGE 2.000E+00,1.000E+00

**Description** This command is valid when the <Parameter> of :MEASure:TRACe<x>:AREA<x>:<Parameter> is not RISE or FALL.

## 5.18 MEASure Group

### **:MEASure:TRACe<x>?**

Function Queries all settings related to the trace.

Syntax :MEASure:TRACe<x>?

<x> = 1 to 8

Example :MEASURE:TRACE1? -> :MEASURE:TRACE1:  
:MEASURE:TRACE1? ->: MEASURE:TRACE1:  
AREA1:BURST:STATE 0;:MEASURE:TRACE1:  
AREA1:CMEAN:STATE 0;:MEASURE:TRACE1:  
AREA1:CMODE 0;COUNT:STATE 0;:  
MEASURE:TRACE1:AREA1:CRMS:STATE 0;:  
MEASURE:TRACE1:AREA1:CSDEVIATION:  
STATE 0;:MEASURE:TRACE1:AREA1:DELAY:  
MEASURE:COUNT 1;POLARITYRISE;:  
MEASURE:TRACE1:AREA1:DELAY:REFERENCE:  
COUNT 1;POLARITY RISE;TRACE 1;:MEASURE:  
TRACE1:AREA1:DELAY:SOURCE TRIGGER;  
STATE 0;:MEASURE:TRACE1:AREA1:  
DPROXIMAL:MODE PERCENT;PERCENT 10,90;  
UNIT -3.0000000E+00,3.0000000E+00;:  
MEASURE:TRACE1:AREA1:DT:STATE 0;:  
MEASURE:TRACE1:AREA1:DUTYCYCLE:  
STATE 0;:MEASURE:TRACE1:AREA1:FALL:  
STATE 0;:MEASURE:TRACE1:AREA1:  
FREQUENCY:STATE 0;:MEASURE:TRACE1:  
AREA1:HIG:STATE 0;:MEASURE:TRACE1:  
AREA1:HILOW:STATE 0;:MEASURE:TRACE1:  
AREA1:LOW:STATE 0;:MEASURE:TRACE1:  
AREA1:MAXIMUM:STATE 0;:MEASURE:TRACE1:  
AREA1:MEAN:STATE 0;:MEASURE:TRACE1:  
AREA1:METHOD AUTO;MINIMUM:STATE 0;:  
MEASURE:TRACE1:AREA1:NOVERSHOOT:  
STATE 0;:MEASURE:TRACE1:AREA1:NWIDTH:  
STATE 0;:MEASURE:TRACE1:AREA1:  
PERFREQUENCY:STATE 0;:MEASURE:TRACE1:  
AREA1:PERIOD:STATE 0;:MEASURE:TRACE1:  
AREA1:POVERSHOOT:STATE 0;:MEASURE:  
TRACE1:AREA1:PTOPEAK:STATE 0;:MEASURE:  
TRACE1:AREA1:PWID:STATE 0;:MEASURE:  
TRACE1:AREA1:RISE:STATE 0;:MEASURE:  
TRACE1:AREA1:RMS:STATE 0 . . . . .

### **:MEASure:TRACe<x>:AREA<x>?**

Function Queries all settings related to the area.

Syntax :MEASure:TRACe<x>:AREA<x>?

<x> of TRACe<x> = 1 to 8

<x> of AREA<x> = 1 or 2

Example :MEASURE:TRACE1:AREA1?  
-> :MEASURE:TRACE1:AREA1:BURST:STATE 0;:  
MEASURE:TRACE1:AREA1:CMEAN:STATE 0;:  
MEASURE:TRACE1:AREA1:CMODE 0;COUNT:  
STATE 0;:MEASURE:TRACE1:AREA1:CRMS:  
STATE 0;:MEASURE:TRACE1:AREA1:  
CSDEVIATION:STATE 0;:MEASURE:TRACE1:  
AREA1:DELAY:MEASURE:COUNT 1;POLARITY  
RISE;:MEASURE:TRACE1:AREA1:DELAY:  
REFERENCE:COUNT 1;POLARITY RISE;  
TRACE1;:MEASURE:TRACE1:AREA1:DELAY:  
SOURCE TRIGGER;STATE 0;:MEASURE:TRACE1:  
AREA1:DPROXIMAL:MODE PERCENT;  
PERCENT 10,90;  
UNIT -3.0000000E+00,3.0000000E+00;:  
MEASURE:TRACE1:AREA1:DT:STATE 0;:  
MEASURE:TRACE1:AREA1:DUTYCYCLE:  
STATE 0;:MEASURE:TRACE1:AREA1:FALL:  
STATE 0;:MEASURE:TRACE1:AREA1:  
FREQUENCY:STATE 0;:MEASURE:TRACE1:  
AREA1:HIG:STATE 0;:MEASURE:TRACE1:  
AREA1:HILOW:STATE 0;:MEASURE:TRACE1:  
AREA1:LOW:STATE 0;:MEASURE:TRACE1:  
AREA1:MAXIMUM:STATE 0;:MEASURE:TRACE1:  
AREA1:MEAN:STATE 0;:MEASURE:TRACE1:  
AREA1:METHOD AUTO;MINIMUM:STATE 0;:  
MEASURE:TRACE1:AREA1:NOVERSHOOT:  
STATE 0;:MEASURE:TRACE1:AREA1:NWIDTH:  
STATE 0;:MEASURE:TRACE1:AREA1:  
PERFREQUENCY:STATE 0;:MEASURE:TRACE1:  
AREA1:PERIOD:STATE 0;:MEASURE:TRACE1:  
AREA1:POVERSHOOT:STATE 0 . . . . .

### **:MEASure:TRACe<x>:AREA<x>:ALL**

Function Turns ON/OFF all waveform parameters.

Syntax :MEASure:TRACe<x>:AREA<x>:ALL

{<Boolean>}

<x> of TRACe<x> = 1 to 8

<x> of AREA<x> = 1 or 2

Example :MEASURE:TRACE1:AREA1:ALL ON



**:MEASure:TRACe<x>:AREA<x>:<Parameter>?**

**Function** Queries all settings related to the waveform parameter.

**Syntax** :MEASure:TRACe<x>:AREA<x>:<Parameter>?

<x> of TRACe<x> = 1 to 8

<x> of AREA<x> = 1 or 2

<Parameter> = {BURSt|CMEan|COUNT|CRMS|CSDeViation|DELaY|DT|DUTYcycle|FALL|FREQuency|HIGH|HILow|LOW|MAXimum|MEAN|MINimum|NOVershoot|NWIDth|PERFrequency|PERiod|POVershoot|PTOPeak|PWIDth|RISE|RMS|SDEViation|TYCInteg|TYINteg|V1|V2}

**Example** (The following is an example for the maximum value of trace 1 and area 1.)

```
:MEASURE:TRACE1:AREA1:MAXIMUM?
-> :MEASURE:TRACE1:AREA1:MAXIMUM:
STATE 0
```

**:MEASure:TRACe<x>:AREA<x>:<Parameter>:COUNT?**

**Function** Queries the continuous statistical processing count of the waveform parameter.

**Syntax** :MEASure:TRACe<x>:AREA<x>:<Parameter>:COUNT?

<x> of TRACe<x> = 1 to 8

<x> of AREA<x> = 1 or 2

<Parameter> = {BURSt|CMEan|COUNT|CRMS|CSDeViation|DELaY|DT|DUTYcycle|FALL|FREQuency|HIGH|HILow|LOW|MAXimum|MEAN|MINimum|NOVershoot|NWIDth|PERFrequency|PERiod|POVershoot|PTOPeak|PWIDth|RISE|RMS|SDEViation|TYCInteg|TYINteg|V1|V2}

**Example** (The following is an example for the maximum value of trace 1 and area 1.)

```
:MEASURE:TRACE1:AREA1:MAXIMUM:COUNT?
-> :MEASURE:TRACE1:AREA1:MAXIMUM:
COUNT 0
```

**:MEASure:TRACe<x>:AREA<x>:<Parameter>:{MAXimum|MEAN|MINimum|SDEViation}?**

**Function** Queries the statistical value of the waveform parameter.

**Syntax** :MEASure:TRACe<x>:AREA<x>:<Parameter>:{MAXimum|MEAN|MINimum|SDEViation}?

<x> of TRACe<x> = 1 to 8

<x> of AREA<x> = 1 or 2

<Parameter> = {BURSt|CMEan|COUNT|CRMS|CSDeViation|DELaY|DT|DUTYcycle|FALL|FREQuency|HIGH|HILow|LOW|MAXimum|MEAN|MINimum|NOVershoot|NWIDth|PERFrequency|PERiod|POVershoot|PTOPeak|PWIDth|RISE|RMS|SDEViation|TYCInteg|TYINteg|V1|V2}

**Example** (The following is an example for the maximum value.)

```
:MEASURE:TRACE1:AREA1:MAXIMUM:
MAXIMUM? -> :MEASURE:TRACE1:AREA1:
MAXIMUM:MAXIMUM 1.000E+00
```

**:MEASure:TRACe<x>:AREA<x>:<Parameter>:STATE**

**Function** Turns ON/OFF the waveform parameter or queries the current setting.

**Syntax** :MEASure:TRACe<x>:AREA<x>:<Parameter>:STATE {<Boolean>}

STATE {<Boolean>}

:MEASure:TRACe<x>:AREA<x>:<Parameter>:STATE?

<x> of TRACe<x> = 1 to 8

<x> of AREA<x> = 1 or 2

<Parameter> = {BURSt|CMEan|COUNT|CRMS|CSDeViation|DELaY|DT|DUTYcycle|FALL|FREQuency|HIGH|HILow|LOW|MAXimum|MEAN|MINimum|NOVershoot|NWIDth|PERFrequency|PERiod|POVershoot|PTOPeak|PWIDth|RISE|RMS|SDEViation|TYCInteg|TYINteg|V1|V2}

**Example** (The following is an example for the maximum value of trace 1 and area 1.)

```
:MEASURE:TRACE1:AREA1:MAXIMUM:STATE ON
:MEASURE:TRACE1:AREA1:MAXIMUM:STATE?
-> :MEASURE:TRACE1:AREA1:MAXIMUM:
STATE 1
```

## 5.18 MEASure Group

### **:MEASure:TRACe<x>:AREA<x>:<Parameter>:VALue?**

- Function** Queries the automated measured value of the waveform parameter.
- Syntax** :MEASure:TRACe<x>:AREA<x>:<Parameter>:VALue? {<NRf>}  
 <x> of TRACe<x> = 1 to 8  
 <x> of AREA<x> = 1 or 2  
 <Parameter> = {BURSt|CMEan|COUNT|CRMS|CSDeViation|DELay|DT|DUTYcycle|FALL|FREQuency|HIGH|HILow|LOW|MAXimum|MEAN|MINimum|NOVershoot|NWIDTH|PERFrequency|PERiod|POVershoot|PTOPeak|PWIDTH|RISE|RMS|SDEViation|TYCInteg|TYINteg|V1|V2}  
 <NRf> = 1 to 100000
- Example** (The following is an example for the maximum value of trace 1 and area 1.)  
 :MEASURE:TRACE1:AREA1:MAXIMUM:VALUE?  
 -> :MEASURE:TRACE1:AREA1:MAXIMUM:VALUE 1.000E+00
- Description**
- If the measurement is not possible, "NAN (Not A Number)" is returned.
  - <NRf> indicates the nth automated measured value in the past.  
 In the case of cycle statistical processing, specify the <NRf> cycle from the left of the screen.  
 To specify the oldest automated measured value, specify 1.  
 If <NRf> is omitted, the latest automated measured value is specified.  
 If the value corresponding to the relevant count is not present, "NAN (Not A Number)" is returned.

### **:MEASure:TRACe<x>:AREA<x>:DELay:MEASure?**

- Function** Queries all settings related to the measurement conditions of the source waveform of the delay measurement between channels.
- Syntax** :MEASure:TRACe<x>:AREA<x>:DELay:MEASure?  
 <x> of TRACe<x> = 1 to 8  
 <x> of AREA<x> = 1 or 2
- Example** :MEASURE:TRACE1:AREA1:DELAY:MEASURE?  
 -> :MEASURE:TRACE1:AREA1:DELAY:MEASURE:COUNT 1;POLARITY RISE

### **:MEASure:TRACe<x>:AREA<x>:DELay:MEASure:COUNT**

- Function** Sets the edge detection count of the source waveform of the delay measurement between channels or queries the current setting.
- Syntax** :MEASure:TRACe<x>:AREA<x>:DELay:MEASure:COUNT {<NRf>}  
 :MEASure:TRACe<x>:AREA<x>:DELay:MEASure:COUNT?  
 <x> of TRACe<x> = 1 to 8  
 <x> of AREA<x> = 1 or 2  
 <NRf> = 1 to 10
- Example** :MEASURE:TRACE1:AREA1:DELAY:MEASURE:COUNT 1  
 :MEASURE:TRACE1:AREA1:DELAY:MEASURE:COUNT? -> :MEASURE:TRACE1:AREA1:DELAY:MEASURE:COUNT 1

### **:MEASure:TRACe<x>:AREA<x>:DELay:MEASure:POLarity**

- Function** Sets the polarity of the source waveform of the delay measurement between channels or queries the current setting.
- Syntax** :MEASure:TRACe<x>:AREA<x>:DELay:MEASure:POLarity {FALL|RISE}  
 :MEASure:TRACe<x>:AREA<x>:DELay:MEASure:POLarity?  
 <x> of TRACe<x> = 1 to 8  
 <x> of AREA<x> = 1 or 2
- Example** :MEASURE:TRACE1:AREA1:DELAY:MEASURE:POLARITY RISE  
 :MEASURE:TRACE1:AREA1:DELAY:MEASURE:POLARITY? -> :MEASURE:TRACE1:AREA1:DELAY:MEASURE:POLARITY RISE

### **:MEASure:TRACe<x>:AREA<x>:DELay:REFERENCE?**

- Function** Queries all settings related to the reference waveform of the delay measurement between channels.
- Syntax** :MEASure:TRACe<x>:AREA<x>:DELay:REFERENCE?  
 <x> of TRACe<x> = 1 to 8  
 <x> of AREA<x> = 1 or 2
- Example** :MEASURE:TRACE1:AREA1:DELAY:REFERENCE?  
 -> :MEASURE:TRACE1:AREA1:DELAY:REFERENCE:COUNT 1;POLARITY FALL;TRACE 1

**:MEASure:TRACe<x>:AREA<x>:DELay:****REFerence:COUNT**

**Function** Sets the edge detection count of the reference waveform of the delay measurement between channels or queries the current setting.

**Syntax**  
 :MEASure:TRACe<x>:AREA<x>:DELay:  
 REFerence:COUNT {<NRF>}  
 :MEASure:TRACe<x>:AREA<x>:DELay:  
 REFerence:COUNT?  
 <x> of TRACe<x> = 1 to 8  
 <x> of AREA<x> = 1 or 2  
 <NRF> = 1 to 10

**Example**  
 :MEASURE:TRACE1:AREA1:DELAY:REFERENCE:  
 COUNT 1  
 :MEASURE:TRACE1:AREA1:DELAY:REFERENCE:  
 COUNT? -> :MEASURE:TRACE1:AREA1:DELAY:  
 REFERENCE:COUNT 1

**:MEASure:TRACe<x>:AREA<x>:DELay:****REFerence:POLarity**

**Function** Sets the polarity of the reference waveform of the delay measurement between channels or queries the current setting.

**Syntax**  
 :MEASure:TRACe<x>:AREA<x>:DELay:  
 REFerence:POLarity {FALL|RISE}  
 :MEASure:TRACe<x>:AREA<x>:DELay:  
 REFerence:POLarity?  
 <x> of TRACe<x> = 1 to 8  
 <x> of AREA<x> = 1 or 2

**Example**  
 :MEASURE:TRACE1:AREA1:DELAY:REFERENCE:  
 POLARITY FALL  
 :MEASURE:TRACE1:AREA1:DELAY:REFERENCE:  
 POLARITY? -> :MEASURE:TRACE1:AREA1:  
 DELAY:REFERENCE:POLARITY FALL

**:MEASure:TRACe<x>:AREA<x>:DELay:****REFerence:TRACe**

**Function** Sets the trace of the reference waveform of the delay measurement between channels or queries the current setting.

**Syntax**  
 :MEASure:TRACe<x>:AREA<x>:DELay:  
 REFerence:TRACe {<NRF>}  
 :MEASure:TRACe<x>:AREA<x>:DELay:  
 REFerence:TRACe?  
 <x> of TRACe<x> = 1 to 8  
 <x> of AREA<x> = 1 or 2  
 <NRF> = 1 to 8

**Example**  
 :MEASURE:TRACE1:AREA1:DELAY:REFERENCE:  
 TRACE 1  
 :MEASURE:TRACE1:AREA1:DELAY:REFERENCE:  
 TRACE? -> :MEASURE:TRACE1:AREA1:DELAY:  
 REFERENCE:TRACE 1

**:MEASure:TRACe<x>:AREA<x>:DELay:****SOURCE**

**Function** Sets the reference of the delay measurement between channels or queries the current setting.

**Syntax**  
 :MEASure:TRACe<x>:AREA<x>:DELay:SOURce  
 {TRACe|TRIGger}  
 :MEASure:TRACe<x>:AREA<x>:DELay:SOURce?  
 <x> of TRACe<x> = 1 to 8  
 <x> of AREA<x> = 1 or 2

**Example**  
 :MEASURE:TRACE1:AREA1:DELAY:  
 SOURCE TRACE  
 :MEASURE:TRACE1:AREA1:DELAY:SOURCE?  
 -> :MEASURE:TRACE1:AREA1:DELAY:  
 SOURCE TRACE

**:MEASure:TRACe<x>:AREA<x>:DPRoximal?**

**Function** Queries all settings related to the distal and proximal values.

**Syntax**  
 :MEASure:TRACe<x>:AREA<x>:DPRoximal?  
 <x> of TRACe<x> = 1 to 8  
 <x> of AREA<x> = 1 or 2

**Example**  
 :MEASURE:TRACE1:AREA1:DPROXIMAL?  
 -> ;MEASURE:TRACE1:AREA1:DPROXIMAL:  
 MODE PERCENT;PERCENT 10,20;UNIT  
 1.000E+00,1.000E+00

**Description** This command is valid when the <Parameter> of  
 :MEASure:TRACe<x>:AREA<x>:<Parameter> is  
 RISE or FALL.

**:MEASure:TRACe<x>:AREA<x>:DPRoximal:  
MODE**

**Function** Sets the unit of the distal and proximal values or queries the current setting.

**Syntax**  
 :MEASure:TRACe<x>:AREA<x>:DPRoximal:  
 MODE {PERCent|UNIT}  
 :MEASure:TRACe<x>:AREA<x>:DPRoximal:  
 MODE?  
 <x> of TRACe<x> = 1 to 8  
 <x> of AREA<x> = 1 or 2

**Example**  
 :MEASURE:TRACE1:AREA1:DPROXIMAL:  
 MODE PERCENT  
 :MEASURE:TRACE1:AREA1:DPROXIMAL:MODE?  
 -> :MEASURE:TRACE1:AREA1:DPROXIMAL:  
 MODE PERCENT

**Description** This command is valid when the <Parameter> of  
 :MEASure:TRACe<x>:AREA<x>:<Parameter> is  
 RISE or FALL.

## 5.18 MEASure Group

### **:MEASure:TRACe<x>:AREA<x>:DPRoximal:PERCent**

**Function** Sets the distal and proximal values as a percentage or queries the current setting.

**Syntax** :MEASure:TRACe<x>:AREA<x>:DPRoximal:PERCent {<NRf>,<NRf>}  
:MEASure:TRACe<x>:AREA<x>:DPRoximal:PERCent?  
<x> of TRACe<x> = 1 to 8  
<x> of AREA<x> = 1 or 2  
<NRf> = 0 to 100 (%)

**Example** :MEASURE:TRACE1:AREA1:DPROXIMAL:PERCENT 10,90  
:MEASURE:TRACE1:AREA1:DPROXIMAL:PERCENT? -> :MEASURE:TRACE1:AREA1:DPROXIMAL:PERCENT 10,90

**Description** This command is valid when the <Parameter> of :MEASure:TRACe<x>:AREA<x>:<Parameter> is RISE or FALL.

### **:MEASure:TRACe<x>:AREA<x>:DPRoximal:UNIT**

**Function** Sets the distal and proximal values in the specified unit or queries the current setting.

**Syntax** :MEASure:TRACe<x>:AREA<x>:DPRoximal:UNIT {(<NRf>,<NRf>)|(<Voltage>,<Voltage>)|(<Current>,<Current>)}  
:MEASure:TRACe<x>:AREA<x>:DPRoximal:UNIT?  
<x> of TRACe<x> = 1 to 8  
<x> of AREA<x> = 1 or 2  
<NRf>, <Voltage>, and <Current> = See the DL9000 User's Manual.

**Example** :MEASURE:TRACE1:AREA1:DPROXIMAL:UNIT 1,-1  
:MEASURE:TRACE1:AREA1:DPROXIMAL:UNIT?  
-> :MEASURE:TRACE1:AREA1:DPROXIMAL:UNIT -1.000E+00,1.000E+00

**Description** This command is valid when the <Parameter> of :MEASure:TRACe<x>:AREA<x>:<Parameter> is RISE or FALL.

### **:MEASure:TRACe<x>:AREA<x>:METHod**

**Function** Sets the method for detecting the High/Low level for automated measurement of waveform parameters or queries the current setting.

**Syntax** :MEASure:TRACe<x>:AREA<x>:METHod {AUTO|HISTogram|MAXMin}  
:MEASure:TRACe<x>:AREA<x>:METHod?  
TRACe<x>: <x> = 1 to 8  
AREA<x>: <x> = 1 or 2

**Example** :MEASURE:TRACE1:AREA1:METHOD AUTO  
:MEASURE:TRACE1:AREA1:METHOD?  
-> :MEASURE:TRACE1:AREA1:METHOD AUTO

### **:MEASure:TRANge<x> (Time Range)**

**Function** Sets the measurement range or queries the current setting.

**Syntax** :MEASure:TRANge<x> {<NRf>,<NRf>}  
:MEASure:TRANge<x>?  
<x> = 1 or 2  
<NRf> = -5 to 5 (div)

**Example** :MEASURE:TRANGE1 -5,0  
:MEASURE:TRANGE1? -> :MEASURE:TRANGE1 0.000E+00,-5.000E+00

### **:MEASure:WAIT?**

**Function** Waits for the completion of the automated measurement with a timeout option.

**Syntax** MEASure:WAIT? {<NRf>}  
<NRf> = 1 to 360000 (timeout value, in units of 10 ms)

**Example** MEASURE:WAIT 100? -> :MEASURE:WAIT 1

**Description** If the execution of the automated measurement completes within the timeout value, 0 is returned; if it is not complete or automated measurement is not being executed, 1 is returned. Even if the timeout value is set long, 0 is returned when the execution of the automated measurement is complete.

### **:MEASure:WINDow<x>**

**Function** Sets the measurement source window of the area or queries the current setting.

**Syntax** :MEASure:WINDow<x> {MAIN|Z1|Z2}  
:MEASure:WINDow<x>?  
<x> = 1 or 2

**Example** :MEASURE:WINDOW1 MAIN  
:MEASURE:WINDOW1?  
-> :MEASURE:WINDOW1 MAIN

## 5.19 REFERENCE GROUP

### **:REfERENCE<x>?**

Function Queries all settings related to the reference.

Syntax :REfERENCE<x>?

<x> = 1 to 4

Example :REFERENCE1? -> :REFERENCE1:  
SELECT REFERENCE;DISPLAY 1;INVERT 0;  
LABEL:DEFINE "REF1";MODE 1;:REFERENCE1:  
POSITION 1.000E+00;SVALUE 1

### **:REfERENCE<x>:DISPlay**

Function Turns ON/OFF the display of the reference or queries the current setting.

Syntax :REfERENCE<x>:DISPlay {<Boolean>}

:REfERENCE<x>:DISPlay?

<x> = 1 to 4

Example :REFERENCE1:DISPLAY ON  
:REFERENCE1:DISPLAY? -> :REFERENCE1:  
DISPLAY 1

### **:REfERENCE<x>:INVert**

Function Sets the inverted display of the reference or queries the current setting.

Syntax :REfERENCE<x>:INVert {<Boolean>}

:REfERENCE<x>:INVert?

<x> = 1 to 4

Example :REFERENCE1:INVERT ON  
:REFERENCE1:INVERT? -> :REFERENCE1:  
INVERT 1

### **:REfERENCE<x>:LABel?**

Function Queries all settings related to the waveform label of the reference.

Syntax :REfERENCE<x>:LABel?

<x> = 1 to 4

Example :REFERENCE1:LABEL? -> :REFERENCE1:  
LABEL:DEFINE "REF1";MODE 1

### **:REfERENCE<x>:LABel[:DEFine]**

Function Sets the waveform label of the reference or queries the current setting.

Syntax :REfERENCE<x>:LABel[:DEFine] {<String>}

:REfERENCE<x>:LABel[:DEFine]?

<x> = 1 to 4

<String> = Up to 8 characters

Example :REFERENCE1:LABEL:DEFINE "REF1"  
:REFERENCE1:LABEL:DEFINE?  
-> :REFERENCE1:LABEL:DEFINE "REF1"

### **:REfERENCE<x>:LABel:MODE**

Function Turns ON/OFF the waveform label display of the reference or queries the current setting.

Syntax :REfERENCE<x>:LABel:MODE {<Boolean>}

:REfERENCE<x>:LABel:MODE?

<x> = 1 to 4

Example :REFERENCE1:LABEL:MODE ON  
:REFERENCE1:LABEL:MODE? -> :REFERENCE1:  
LABEL:MODE 1

### **:REfERENCE<x>:LOAD**

Function Loads the waveform to the reference.

Syntax :REfERENCE<x>:LOAD {<Nrf>}

<x> = 1 to 4

<Nrf> = 1 to 12 (1 to 8 are traces and 9 to 12 are internal memories)

Example :REFERENCE1:LOAD 1

### **:REfERENCE<x>:POSition**

Function Sets the vertical position of the reference or queries the current setting.

Syntax :REfERENCE<x>:POSition {<Nrf>}

:REfERENCE<x>:POSition?

<x> = 1 to 4

<Nrf> = -4 to 4 (div)

Example :REFERENCE1:POSITION 1  
:REFERENCE1:POSITION? -> :REFERENCE1:  
POSITION 1.000E+00

### **:REfERENCE<x>:SElect**

Function Sets the waveform (computation or reference) to the computation channel or queries the current setting.

Syntax :REfERENCE<x>:SElect {MATH|REFERENCE}

:REfERENCE<x>:SElect?

<x> = 1 to 4

Example :REFERENCE1:SELECT MATH  
:REFERENCE1:SELECT? -> :REFERENCE1:  
SELECT MATH

### **:REfERENCE<x>:SVALue (Scale VALUE)**

Function Turns ON/OFF the scale display of the reference or queries the current setting.

Syntax :REfERENCE<x>:SVALue {<Boolean>}

:REfERENCE<x>:SVALue?

<x> = 1 to 4

Example :REFERENCE1:SVALUE ON  
:REFERENCE1:SVALUE? -> :REFERENCE1:  
SVALUE 1

## 5.20 SEARCh Group

### : SEARCh<x>?

**Function** Queries all settings related to the search function.

**Syntax** :SEARCh<x>?  
<x> = 1 or 2

**Example** :SEARCH1? -> :SEARCH1:CLOCK:SOURCE 1;  
POLARITY FALL;:SEARCH1:DECIMATION 1;  
HOLDOFF 0.000E+00;LOGIC AND;  
POLARITY RISE;SMODE OFF;SPATTERN:CLOCK:  
MODE 1;POLARITY FALL;SOURCE 1;:SEARCH1:  
SPATTERN:CS 1;LATCH:TRACE 1;  
POLARITY FALL;:SEARCH1:SPATTERN:SETUP:  
BITRATE 1.000E+00;  
PATTERN "1100110111101111";:SEARCH1:  
SPOINT -5.000E+00;STRACE 1;TRACE1:  
CONDITION DONTCARE;  
HYSTERESIS 1.000E+00;LEVEL 0.000E+00;;  
SEARCH1:TRACE2:CONDITION DONTCARE;  
HYSTERESIS 1.000E+00;LEVEL 0.000E+00;;  
SEARCH1:TRACE3:CONDITION DONTCARE;  
HYSTERESIS 1.000E+00;LEVEL 0.000E+00;;  
SEARCH1:TRACE4:CONDITION DONTCARE;  
HYSTERESIS 1.000E+00;LEVEL 0.000E+00;;  
SEARCH1:TRACE5:CONDITION DONTCARE;  
HYSTERESIS 1.000E+00;LEVEL 0.000E+00;;  
SEARCH1:TRACE6:CONDITION DONTCARE;  
HYSTERESIS 1.000E+00;LEVEL 0.000E+00;;  
SEARCH1:TRACE7:CONDITION DONTCARE;  
HYSTERESIS 1.000E+00;LEVEL 0.000E+00;;  
SEARCH1:TRACE8:CONDITION DONTCARE;  
HYSTERESIS 1.000E+00;LEVEL 0.000E+00;;  
SEARCH1:TYPE EDGE;WIDTH:MODE OUT;  
TIME1 1.000E+00;TIME2 2.000E+00;  
TYPE PULSE

### : SEARCh<x>: ABORt

**Function** Aborts the search.

**Syntax** :SEARCh<x>:ABORt  
<x> = 1 or 2

**Example** :SEARCH1:ABORt

### : SEARCh<x>: CLOCk?

**Function** Queries all settings related to the clock channel.

**Syntax** :SEARCh<x>:CLOCk?  
<x> = 1 or 2

**Example** :SEARCH1:CLOCK? -> :SEARCH1:CLOCK:  
SOURCE 1;POLARITY RISE

### : SEARCh<x>: CLOCk: POLARity

**Function** Sets the polarity of the clock channel or queries the current setting.

**Syntax** :SEARCh<x>:CLOCk:POLARity {FALL|RISE}  
:SEARCh<x>:CLOCk:POLARity?  
<x> = 1 or 2

**Example** :SEARCH1:CLOCK:POLARITY RISE  
:SEARCH1:CLOCK:POLARITY? -> :SEARCH1:  
CLOCK:POLARITY RISE

**Description**• This command is invalid when :SEARCh<x>:  
CLOCk:SOURce NONE.

- This command is valid when :SEARCh<x>:TYPE STATE.
- This command is valid when :SEARCh<x>:TYPE WIDTH and :SEARCh<x>:WIDTH:TYPE PStAte.

### : SEARCh<x>: CLOCk: SOURce

**Function** Sets the clock trace of the search or queries the current setting.

**Syntax** :SEARCh<x>:CLOCk:SOURce {<NRf>|NONE}  
:SEARCh<x>:CLOCk:SOURce?  
<x> = 1 or 2  
<NRf> = 1 to 8

**Example** :SEARCH1:CLOCK:SOURCE NONE  
:SEARCH1:CLOCK:SOURCE? -> :SEARCH1:  
CLOCK:SOURCE NONE

**Description**• This command is valid when :SEARCh<x>:TYPE STATE.

- This command is valid when :SEARCh<x>:TYPE WIDTH and :SEARCh<x>:WIDTH:TYPE PStAte.

### : SEARCh<x>: DECImation

**Function** Sets the decimation detection of the skip mode or queries the current setting.

**Syntax** :SEARCh<x>:DECImation {<NRf>}  
:SEARCh<x>:DECImation?  
<x> = 1 or 2  
<NRf> = 1 to 9999

**Example** :SEARCH1:DECIMATION 1  
:SEARCH1:DECIMATION? -> :SEARCH1:  
DECIMATION 1

**:SEARCH<x>:EXECute**

Function Executes the search. This is an overlap command.

Syntax :SEARCH<x>:EXECute  
<x> = 1 or 2

Example :SEARCH1:EXECUTE

**:SEARCH<x>:HOLDoff**

Function Sets the hold off detection or queries the current setting.

Syntax :SEARCH<x>:HOLDoff {<Time>}  
:SEARCH<x>:HOLDoff?  
<x> = 1 or 2  
<Time> = 0 s to 1 s in 100-ps steps

Example :SEARCH1:HOLDOFF 0S  
:SEARCH1:HOLDOFF? -> :SEARCH1:  
HOLDOFF 0.000E+00

**:SEARCH<x>:LOGic**

Function Sets the search logic or queries the current setting.

Syntax :SEARCH<x>:LOGic {AND|OR}  
:SEARCH<x>:LOGic?  
<x> = 1 or 2

Example :SEARCH1:LOGIC OR  
:SEARCH1:LOGIC? -> :SEARCH1:LOGIC OR

Description • This command is valid when :SEARCH<x>:TYPE  
EQUalify|SPATtern|STATe.

- This command is valid when :SEARCH<x>:  
TYPE WIDTH and :SEARCH<x>:WIDTH:TYPE  
PQUalify|PSTATe.

**:SEARCH<x>:POLarity**

Function Sets the search polarity or queries the current setting.

Syntax :SEARCH<x>:POLarity {ENTER|EXIT|FALL|  
FALSE|NEGative|POSitive|RISE|  
TRUE}  
:SEARCH<x>:POLarity?  
<x> = 1 or 2

Example :SEARCH1:POLARITY ENTER  
:SEARCH1:POLARITY? -> :SEARCH1:  
POLARITY ENTER

Description • {FALL|RISE} is valid when :SEARCH<x>:TYPE  
EDGE|EQUalify.

- {ENTER|EXIT} is valid when :SEARCH<x>:TYPE  
STATE.
- {NEGative|POSitive} is valid when :SEARCH<x>:  
TYPE WIDTH and :SEARCH<x>:WIDTH:TYPE  
PQUalify|PULSe.
- {FALSE|TRUE} is valid when :SEARCH<x>:TYPE  
WIDTH and :SEARCH<x>:WIDTH:TYPE PSTATe.

**:SEARCH<x>:SELEct**

Function Sets the detection waveform number of the search function and queries the position that corresponds to the detection waveform number.

Syntax :SEARCH<x>:SELEct {<NRf>|MAXimum}  
:SEARCH<x>:SELEct?  
<x> = 1 or 2  
<NRf> = 0 to 4999

Example :SEARCH1:SELEct 1  
:SEARCH1:SELEct? -> :SEARCH1:  
SELEct 1.500E+00

Description If there is no searched position, "NAN" is returned for the query.

**:SEARCH<x>:SELEct? MAXimum**

Function Queries the detection count of the search function.

Syntax :SEARCH<x>:SELEct? {MAXimum}  
<x> = 1 or 2

Example :SEARCH1:SELEct? MAXIMUM -> :SEARCH1:  
SELEct 100

Description If there is no searched position, "NAN" is returned for the query.

**:SEARCH<x>:SMODE**

Function Sets the skip mode or queries the current setting.

Syntax :SEARCH<x>:SMODE {DECimation|HOLDoff|  
OFF}  
:SEARCH<x>:SMODE?  
<x> = 1 or 2

Example :SEARCH1:SMODE HOLDOFF  
:SEARCH1:SMODE? -> :SEARCH1:  
SMODE HOLDOFF

**:SEARCH<x>:SPATtern? (Serial Pattern)**

Function Queries all settings related to the serial pattern search.

Syntax :SEARCH<x>:SPATtern?  
<x> = 1 or 2

Example :SEARCH1:SPATtern? -> :SEARCH1:  
SPATtern:CLOCK:MODE 1;POLARITY FALL;  
SOURCE 1;:SEARCH1:SPATtern:CS 1;LATCH:  
TRACE 1;POLARITY FALL;:SEARCH1:  
SPATtern:SETUP:BITRATE 1.000E+00;  
PATTERN "1100110111101111"

**:SEARCH<x>:SPATtern:CLOCK?**

Function Queries all settings related to clock of the serial pattern search.

Syntax :SEARCH<x>:SPATtern:CLOCK?  
<x> = 1 or 2

Example :SEARCH1:SPATtern:CLOCK? -> :SEARCH1:  
SPATtern:CLOCK:MODE 1;  
POLARITY FALL;SOURCE 1

## 5.20 SEARCh Group

### **:SEARCh<x>:SPATtern:CLOCK:MODE**

Function Enables/Disables the clock of the serial pattern search or queries the current setting.

Syntax :SEARCh<x>:SPATtern:CLOCK:  
MODE {<Boolean>}  
:SEARCh<x>:SPATtern:CLOCK:MODE?  
<x> = 1 or 2

Example :SEARCH1:SPATTERN:CLOCK:MODE ON  
:SEARCH1:SPATTERN:CLOCK:MODE?  
-> :SEARCH1:SPATTERN:CLOCK:MODE 1

### **:SEARCh<x>:SPATtern:CLOCK:POLarity**

Function Sets the polarity of the clock trace of the serial pattern search or queries the current setting.

Syntax :SEARCh<x>:SPATtern:CLOCK:  
POLarity {FALL|RISE}  
:SEARCh<x>:SPATtern:CLOCK:POLarity?  
<x> = 1 or 2

Example :SEARCH1:SPATTERN:CLOCK:POLARITY FALL  
:SEARCH1:SPATTERN:CLOCK:POLARITY?  
-> :SEARCH1:SPATTERN:CLOCK:  
POLARITY FALL

Description This command is valid when :SEARCh<x>:  
SPATtern:CLOCK:MODE ON.

### **:SEARCh<x>:SPATtern:CLOCK:SOURce**

Function Sets the clock trace of the serial pattern search or queries the current setting.

Syntax :SEARCh<x>:SPATtern:CLOCK:  
SOURce {<Nrf>}  
:SEARCh<x>:SPATtern:CLOCK:SOURce?  
<x> = 1 or 2  
<Nrf> = 1 to 8

Example :SEARCH1:SPATTERN:CLOCK:SOURCE 1  
:SEARCH1:SPATTERN:CLOCK:SOURCE?  
-> :SEARCH1:SPATTERN:CLOCK:SOURCE 1

Description This command is valid when :SEARCh<x>:  
SPATtern:CLOCK:MODE ON.

### **:SEARCh<x>:SPATtern:CS**

Function Enables/Disables the chip select of the serial pattern search or queries the current setting.

Syntax :SEARCh<x>:SPATtern:CS {<Boolean>}  
:SEARCh<x>:SPATtern:CS?  
<x> = 1 or 2

Example :SEARCH1:SPATTERN:CS ON  
:SEARCH1:SPATTERN:CS? -> :SEARCH1:  
SPATTERN:CS 1

Description This command is valid when :SEARCh<x>:  
SPATtern:CLOCK:MODE ON.

### **:SEARCh<x>:SPATtern:LATCh?**

Function Queries all settings related to latch of the serial pattern search.

Syntax :SEARCh<x>:SPATtern:LATCh?  
<x> = 1 or 2

Example :SEARCH1:SPATTERN:LATCH? -> :SEARCH1:  
SPATTERN:LATCH:TRACE 1;POLARITY FALL

### **:SEARCh<x>:SPATtern:LATCh:POLarity**

Function Sets the polarity of the latch trace of the serial pattern search or queries the current setting.

Syntax :SEARCh<x>:SPATtern:LATCh:  
POLarity {FALL|RISE}  
:SEARCh<x>:SPATtern:LATCh:POLarity?  
<x> = 1 or 2

Example :SEARCH1:SPATTERN:LATCH:POLARITY FALL  
:SEARCH1:SPATTERN:LATCH:POLARITY?  
-> :SEARCH1:SPATTERN:LATCH:  
POLARITY FALL

Description • This command is valid when :SEARCh<x>:  
SPATtern:CLOCK:MODE ON.  
• This command is invalid when :SEARCh<x>:  
SPATtern:LATCh:TRACE NONE.

### **:SEARCh<x>:SPATtern:LATCh:TRACe**

Function Sets the latch trace of the serial pattern search or queries the current setting.

Syntax :SEARCh<x>:SPATtern:LATCh:TRACe {<Nrf>|  
NONE}  
:SEARCh<x>:SPATtern:LATCh:TRACe?  
<x> = 1 or 2  
<Nrf> = 1 to 8

Example :SEARCH1:SPATTERN:LATCH:TRACE 1  
:SEARCH1:SPATTERN:LATCH:TRACE?  
-> :SEARCH1:SPATTERN:LATCH:TRACE 1

Description This command is valid when :SEARCh<x>:  
SPATtern:CLOCK:MODE ON.

### **:SEARCh<x>:SPATtern:SETup?**

Function Queries all settings related to setup of the serial pattern search.

Syntax :SEARCh<x>:SPATtern:SETup?  
<x> = 1 or 2

Example :SEARCH1:SPATTERN:SETUP? -> :SEARCH1:  
SPATTERN:SETUP:BITRATE 1.000E+03;DATA:  
ACTIVE HIGH;TRACE 1;:SEARCH1:SPATTERN:  
SETUP:PATTERN "1100"



**: SEARCh<x>: SPATtern [ : SETUp] : BITRate**

Function Sets the bit rate of the serial pattern search or queries the current setting.

Syntax :SEARCh<x>:SPATtern[:SETUp]:  
BITRate {<NRf>}  
:SEARCh<x>:SPATtern[:SETUp]:BITRate?  
<x> = 1 or 2  
<NRf> = 1 to 1G (bps)

Example :SEARCH1:SPATTERN:SETUP:BITRATE 1  
:SEARCH1:SPATTERN:SETUP:BITRATE?  
-> :SEARCH1:SPATTERN:SETUP:  
BITRATE 1.000E+00

Description This command is valid when :SEARCh<x>:  
SPATtern:CLOCK:MODE OFF.

**: SEARCh<x>: SPATtern [ : SETUp] : CLear**

Function Clears the entire pattern of the serial pattern search (to don't care).

Syntax :SEARCh<x>:SPATtern[:SETUp]:CLear  
<x> = 1 or 2

Example :SEARCH1:SPATTERN:SETUP:CLEAR

**: SEARCh<x>: SPATtern [ : SETUp] : DATA?**

Function Queries all settings related to data of the serial pattern search.

Syntax :SEARCh<x>:SPATtern[:SETUp]:DATA?  
<x> = 1 or 2

Example :SEARCH1:SPATTERN:SETUP:DATA?  
-> :SEARCH1:SPATTERN:SETUP:DATA:  
ACTIVE HIGH;TRACE 1

**: SEARCh<x>: SPATtern [ : SETUp] : DATA:****ACTive**

Function Sets the active level of the data trace of the serial pattern search or queries the current setting.

Syntax :SEARCh<x>:SPATtern[:SETUp]:DATA:  
ACTive {HIGH|LOW}  
:SEARCh<x>:SPATtern[:SETUp]:DATA:  
ACTive?  
<x> = 1 or 2

Example :SEARCH1:SPATTERN:SETUP:DATA:  
ACTIVE HIGH  
:SEARCH1:SPATTERN:SETUP:DATA:ACTIVE?  
-> :SEARCH1:SPATTERN:SETUP:DATA:  
ACTIVE HIGH

**: SEARCh<x>: SPATtern [ : SETUp] : DATA: TRACe**

Function Sets the data trace of the serial pattern search or queries the current setting.

Syntax :SEARCh<x>:SPATtern[:SETUp]:DATA:  
TRACe {<NRf>}  
:SEARCh<x>:SPATtern[:SETUp]:DATA:TRACe?  
<x> = 1 or 2  
<NRf> = 1 to 8

Example :SEARCH1:SPATTERN:SETUP:DATA:TRACE 1  
:SEARCH1:SPATTERN:SETUP:DATA:TRACE?  
-> :SEARCH1:SPATTERN:SETUP:DATA:TRACE 1

**: SEARCh<x>: SPATtern [ : SETUp] : HEXA**

Function Sets the pattern of the serial pattern search in hexadecimal notation.

Syntax :SEARCh<x>:SPATtern[:SETUp]:HEXA  
{<String>}  
<x> = 1 or 2  
<String> = Up to 32 characters by combining '0' to 'F' and 'X'

Example :SEARCH1:SPATTERN:SETUP:HEXA "ABCD"

**: SEARCh<x>: SPATtern [ : SETUp] : PATtern**

Function Sets the pattern of the serial pattern search in binary notation or queries the current setting.

Syntax :SEARCh<x>:SPATtern[:SETUp]:  
PATtern {<String>}  
:SEARCh<x>:SPATtern[:SETUp]:PATtern?  
<x> = 1 or 2  
<String> = Up to 128 characters by combining '0', '1', and 'X'

Example :SEARCH1:SPATTERN:SETUP:  
PATTERN "1100110111101111"  
:SEARCH1:SPATTERN:SETUP:PATTERN?  
-> :SEARCH1:SPATTERN:SETUP:  
PATTERN "1100110111101111"

**: SEARCh<x>: SPOint**

Function Sets the search start position or queries the current setting.

Syntax :SEARCh<x>:SPOint {<NRf>}  
:SEARCh<x>:SPOint?  
<x> = 1 or 2  
<NRf> = -5 to 5 (div)

Example :SEARCH1:SPOINT 1  
:SEARCH1:SPOINT?  
-> :SEARCH1:SPOINT 1.000E+00

**: SEARCh<x>: STRace**

Function Sets the search source trace or queries the current setting.

Syntax :SEARCh<x>:STRace {<NRf>}  
:SEARCh<x>:STRace?  
<x> = 1 or 2  
<NRf> = 1 to 8

Example :SEARCH1:STRACE 1  
:SEARCH1:STRACE? -> :SEARCH1:STRACE 1

Description • This command is valid when :SEARCh<x>:TYPE  
EDGE|EQualify.  
• This command is valid when :SEARCh<x>:  
TYPE WIDTH and :SEARCh<x>:WIDTH:TYPE  
PQQualify|PULSe.

## 5.20 SEARCh Group

### **:SEARCh<x>:TRACe<x>?**

**Function** Queries all settings related to the search conditions of the trace.

**Syntax** :SEARCh<x>:TRACe<x>?  
<x> of SEARCh<x> = 1 or 2  
<x> of TRACe<x> = 1 to 8

**Example** :SEARCH1:TRACE1? -> :SEARCH1:TRACE1:  
CONDITION DONTCARE;  
HYSTERESIS 1.000E+00;LEVEL 0.000E+00

### **:SEARCh<x>:TRACe<x>:CONDition**

**Function** Sets the condition to be satisfied for the trace or queries the current setting.

**Syntax** :SEARCh<x>:TRACe<x>:  
CONDition {DONTcare|HIGH|LOW}  
:SEARCh<x>:TRACe<x>:CONDition?  
<x> of SEARCh<x> = 1 or 2  
<x> of TRACe<x> = 1 to 8

**Example** :SEARCH1:TRACE1:CONDITION HIGH  
:SEARCH1:TRACE1:CONDITION? -> :SEARCH1:  
TRACE1:CONDITION HIGH

**Description** • This command is valid when :SEARCh<x>:TYPE  
EQUalify|SPATtern|STATe.  
• This command is valid when :SEARCh<x>:  
TYPE WIDTH and :SEARCh<x>:WIDTH:TYPE  
PQUalify|PSTATe.

### **:SEARCh<x>:TRACe<x>:HYSTeresis**

**Function** Sets the hysteresis of the trace or queries the current setting.

**Syntax** :SEARCh<x>:TRACe<x>:HYSTeresis {<NRF>}  
:SEARCh<x>:TRACe<x>:HYSTeresis?  
<x> of SEARCh<x> = 1 or 2  
<x> of TRACe<x> = 1 to 8  
<NRF> = 0 to 4 (div, in 0.1-div steps)

**Example** :SEARCH1:TRACE1:HYSTERESIS 1  
:SEARCH1:TRACE1:HYSTERESIS?  
-> :SEARCH1:TRACE1:HYSTERESIS 1.000E+00

### **:SEARCh<x>:TRACe<x>:LEVel**

**Function** Sets the threshold level of the trace or queries the current setting.

**Syntax** :SEARCh<x>:TRACe<x>:LEVel {<NRF>|  
<Voltage>|<Current>}  
:SEARCh<x>:TRACe<x>:LEVel?  
<x> of SEARCh<x> = 1 or 2  
<x> of TRACe<x> = 1 to 8  
<NRF>, <Voltage>, and <Current> = See the DL9000  
User's Manual.

**Example** :SEARCH1:TRACE1:LEVEL 0  
:SEARCH1:TRACE1:LEVEL? -> :SEARCH1:  
TRACE1:LEVEL 0.000E+00

### **:SEARCh<x>:TYPE**

**Function** Sets the search type or queries the current setting.

**Syntax** :SEARCh<x>:TYPE {EDGE|EQUalify|  
SPATtern|STATe|WIDTH}  
:SEARCh<x>:TYPE?  
<x> = 1 or 2

**Example** :SEARCH1:TYPE EDGE  
:SEARCH1:TYPE? -> :SEARCH1:TYPE EDGE

### **:SEARCh<x>:WIDTh?**

**Function** Queries all settings related to the pulse width search.

**Syntax** :SEARCh<x>:WIDTh?  
<x> = 1 or 2

**Example** :SEARCH1:WIDTH? -> :SEARCH1:WIDTH:  
MODE OUT;TIME1 1.000E-09;  
TIME2 2.000E-09;TYPE PULSE

### **:SEARCh<x>:WIDTh:MODE**

**Function** Sets the pulse width determination mode or queries the current setting.

**Syntax** :SEARCh<x>:WIDTh:MODE {BETWeen|IN|NOTBe  
tween|OUT|TIMEout}  
:SEARCh<x>:WIDTh:MODE?  
<x> = 1 or 2

**Example** :SEARCH1:WIDTH:MODE TIMEOUT  
:SEARCH1:WIDTH:MODE? -> :SEARCH1:WIDTH:  
MODE TIMEOUT

### **:SEARCh<x>:WIDTh:TIME<x>**

**Function** Sets the pulse width of the pulse width search or queries the current setting.

**Syntax** :SEARCh<x>:WIDTh:TIME<x> {<Time>}  
:SEARCh<x>:WIDTh:TIME<x>?  
<x> of SEARCh<x> = 1 or 2  
<x> of TIME<x> = 1 or 2  
<Time> = 1 ns to 10 s in 500-ps steps

**Example** :SEARCH1:WIDTH:TIME1 1S  
:SEARCH1:WIDTH:TIME1? -> :SEARCH1:  
WIDTH:TIME1 1.000E+00

**Description** TIME2 is valid when :SEARCh<x>:WIDTh:MODE  
BETWeen|NOTBetween.

### **:SEARCh<x>:WIDTh:TYPE**

**Function** Sets the pulse width search type or queries the current setting.

**Syntax** :SEARCh<x>:WIDTh:TYPE {PQUalify|PSTATe|  
PULSe}  
:SEARCh<x>:WIDTh:TYPE?  
<x> = 1 or 2

**Example** :SEARCH1:WIDTH:TYPE PQUALIFY  
:SEARCH1:WIDTH:TYPE? -> :SEARCH1:WIDTH:  
TYPE PQUALIFY

---

## 5.21 SNAP Group

**: SNAP**

Function Executes the snapshot.

Syntax : SNAP

Example : SNAP

---

## 5.22 SStart Group

### :SStart?

**Function** Starts the waveform acquisition with the trigger mode set to single. If the waveform acquisition stops within the specified time period, 0 is returned at that point. If not, 1 is returned.

**Syntax** :SStart? {<NRf>}  
<NRf> = 1 to 360000 (10 ms resolution: wait period, START and wait)  
0 (START only. No wait.)  
-360000 to -1 (10 ms unit: wait time, do not START and wait)

**Example** :SSTART? 100 -> :SSTART 0

**Description**

- If the specified time period is positive, data acquisition is started in the SINGLE TRIGGER mode and waits for the operation to stop.
- If the specified time period is 0, data acquisition is started and 0 is returned without waiting for the operation to stop.
- If the specified time period is negative, the instrument simply waits for the operation to stop. Data acquisition is not started.

---

## 5.23 START Group

### : START

Function Starts the waveform acquisition.

Syntax : START

Example : START

Description Use STOP to stop the waveform acquisition.

## 5.24 STATUS Group

The commands in the STATUS group are used to make settings and inquiries related to the communication status function. There are no front panel keys that correspond to the commands in this group. For a description of the status report, see chapter 6.

### **:STATUS?**

**Function** Queries all settings related to the communication status function.

**Syntax** :STATUS?

**Example** :STATUS? -> :STATUS:EESE 0;  
FILTER1 NEVER;FILTER2 NEVER;  
FILTER3 NEVER;FILTER4 NEVER;  
FILTER5 NEVER;FILTER6 NEVER;  
FILTER7 NEVER;FILTER8 NEVER;  
FILTER9 NEVER;FILTER10 NEVER;  
FILTER11 NEVER;FILTER12 NEVER;  
FILTER13 NEVER;FILTER14 NEVER;  
FILTER15 NEVER;FILTER16 NEVER;  
QENABLE 1;QMESSAGE 1

### **:STATUS:CONDITION?**

**Function** Queries the contents of the condition register.

**Syntax** :STATUS:CONDITION?

**Example** :STATUS:CONDITION -> 16

**Description** For details on the condition register, see chapter 6, "Status Report."

### **:STATUS:EESE**

**Function** Sets the extended event enable register or queries the current setting.

**Syntax** :STATUS:EESE <Register>  
:STATUS:EESE?  
<Register> = 0 to 65535

**Example** :STATUS:EESE 257  
:STATUS:EESE? -> :STATUS:EESE 257

**Description** For details on the extended event enable register, see chapter 6, "Status Report."

### **:STATUS:EESR?**

**Function** Queries the content of the extended event register and clears the register.

**Syntax** :STATUS:EESR?

**Example** :STATUS:EESR? -> 1

**Description** For details on the extended event register, see chapter 6, "Status Report."

### **:STATUS:ERROR?**

**Function** Queries the error code and message information (top of the error queue).

**Syntax** :STATUS:ERROR?

**Example** :STATUS:ERROR?  
-> 113,"Undefined header"

### **:STATUS:FILTER<x>**

**Function** Sets the transition filter or queries the current setting.

**Syntax** :STATUS:FILTER<x> {RISE|FALL|BOTH|NEVER}  
:STATUS:FILTER<x>?  
<x> = 1 to 16

**Example** :STATUS:FILTER2 RISE  
:STATUS:FILTER2? -> :STATUS:FILTER2 RISE

**Description** For details on the transition filter, see chapter 6, "Status Report."

### **:STATUS:QENABLE**

**Function** Sets whether to store messages other than errors to the error queue or queries the current setting.

**Syntax** :STATUS:QENABLE {<Boolean>}  
:STATUS:QENABLE?

**Example** :STATUS:QENABLE ON  
:STATUS:QENABLE? -> :STATUS:QENABLE 1

### **:STATUS:QMESSAGE**

**Function** Sets whether or not to attach message information to the response to the "STATUS:ERROR?" query or queries the current setting.

**Syntax** :STATUS:QMESSAGE {<Boolean>}  
:STATUS:QMESSAGE?

**Example** :STATUS:QMESSAGE OFF  
:STATUS:QMESSAGE? -> :STATUS:QMESSAGE 0

### **:STATUS:SPOLL? (Serial Poll)**

**Function** Executes serial polling.

**Syntax** :STATUS:SPOLL?

**Example** :STATUS:SPOLL? -> STATUS:SPOLL 0

**Description** This command is dedicated to the Ethernet interface (option).

---

## 5.25 STOP Group

**: STOP**

Function Stops the waveform acquisition.

Syntax : STOP

Example : STOP

Description Use START to start the waveform acquisition.

## 5.26 SYSTem Group

### **: SYSTem?**

Function Queries all settings related to the system.  
Syntax :SYSTem?  
Example :SYSTEM? -> :SYSTEM:CLICK 1;CLOCK:  
DTIME "2005/05/06",  
"11:37:32", "09:00";MODE 1;:SYSTEM:  
LANGUAGE JAPANESE;MFSIZE SMALL;  
MLANGUAGE ENGLISH;USBKEYBOARD ENGLISH

### **: SYSTem: CLICk**

Function Turns ON/OFF the click sound or queries the current setting.  
Syntax :SYSTem:CLICk {<Boolean>}  
:SYSTem:CLICk?  
Example :SYSTEM:CLICK ON  
:SYSTEM:CLICK? -> :SYSTEM:CLICK 1

### **: SYSTem: CLOCk?**

Function Queries all settings related to the date, time, and time difference with respect to GMT.  
Syntax :SYSTem:CLOCk?  
Example :SYSTEM:CLOCK? -> :SYSTEM:CLOCK:  
DTIME "2005/05/06", "11:37:32", "09:00";  
MODE 1

### **: SYSTem: CLOCk : DTIme**

Function Sets the date, time, and time difference with respect to GMT or queries the current setting.  
Syntax :SYSTem:CLOCk:DTIme {<String>,<String>,<String>}  
:SYSTem:CLOCk:DTIme?  
The left <String> = YYYY/MM/DD. See the DL9000 User's Manual.  
The center <String> = HH:MM:SS. See the DL9000 User's Manual.  
The right <String> = HH:MM. See the DL9000 User's Manual.  
Example :SYSTEM:CLOCK:DTIME "2005/05/06",  
"11:37:32", "09:00"  
:SYSTEM:CLOCK:DTIME? -> :SYSTEM:CLOCK:  
DTIME "2005/05/06",  
"11:37:32", "09:00"

### **: SYSTem: CLOCk : MODe**

Function Turns ON/OFF the date, time, and time difference with respect to GMT or queries the current setting.  
Syntax :SYSTem:CLOCk:MODe {<Boolean>}  
:SYSTem:CLOCk:MODe?  
Example :SYSTEM:CLOCK:MODE ON  
:SYSTEM:CLOCK:MODE? -> :SYSTEM:CLOCK:  
MODE 1

### **: SYSTem: FORMat: IMEMory [ : EXECute]**

Function Formats the internal memory.  
Syntax :SYSTem:FORMat:IMEMory[:EXECute]  
Example :SYSTEM:FORMAT:IMEMORY:EXECUTE

### **: SYSTem: FORMat: IHDD [ : EXECute]**

Function Formats the internal hard disk.  
Syntax :SYSTem:FORMat:IHDD[:EXECute]  
Example :SYSTEM:FORMAT:IHDD:EXECUTE

### **: SYSTem: FORMat: SDElete [ : EXECute] (Sure Delete)**

Function Clears and formats the internal memory.  
Syntax :SYSTem:FORMat:SDElete[:EXECute]  
Example :SYSTEM:FORMAT:SDELETE:EXECUTE

### **: SYSTem: LANGUage**

Function Sets the message language or queries the current setting.  
Syntax :SYSTem:LANGUage {CHINese|ENGLish|  
JAPANese|KOREan}  
:SYSTem:LANGUage?  
Example :SYSTEM:LANGUAGE JAPANESE  
:SYSTEM:LANGUAGE? -> :SYSTEM:  
LANGUAGE JAPANESE

### **: SYSTem: MFSize**

Function Sets the menu font size or queries the current setting.  
Syntax SYSTem:MFSize {LARGE|SMALL}  
Example :SYSTEM:MFSIZE LARGE  
:SYSTEM:MFSIZE? -> :SYSTEM:MFSIZE LARGE

### **: SYSTem: MLANGUage**

Function Sets the menu language or queries the current setting.  
Syntax :SYSTem:MLANGUage {CHINese|ENGLish|JAPA  
Nese|KOREan}  
:SYSTem:MLANGUage?  
Example :SYSTEM:MLANGUAGE ENGLISH  
:SYSTEM:MLANGUAGE? -> :SYSTEM:  
MLANGUAGE ENGLISH

### **: SYSTem: OVERview**

Function Displays system information.  
Syntax :SYSTem:OVERview  
Example :SYSTEM:OVERVIEW

### **: SYSTem: USBKeyboard**

Function Sets the USB keyboard type or queries the current setting.  
Syntax :SYSTem:USBKeyboard {ENGLish|JAPANese}  
:SYSTem:USBKeyboard?  
Example :SYSTEM:USBKEYBOARD ENGLISH  
:SYSTEM:USBKEYBOARD? -> :SYSTEM:  
USBKEYBOARD ENGLISH



## 5.27 TELecomtest Group

### :TELecomtest?

**Function** Queries all settings related to the telecom test.

**Syntax** :TELecomtest?

**Example** :TELECOMTEST? -> :TELECOMTEST:  
CATEGORY MASK;DISPLAY 0;EYEPATTERN:  
DBERATE:STATE 0;:TELECOMTEST:  
EYEPATTERN:EHEIGHT:STATE 0;:  
TELECOMTEST:EYEPATTERN:EWIDTH:  
STATE 0;:TELECOMTEST:EYEPATTERN:FALL:  
STATE 0;:TELECOMTEST:EYEPATTERN:  
JITTER1:STATE 0;:TELECOMTEST:  
EYEPATTERN:JITTER2:STATE 1;:  
TELECOMTEST:EYEPATTERN:PCROSSING:  
STATE 0;:TELECOMTEST:EYEPATTERN:  
PDUTYCYCLE:STATE 0;:TELECOMTEST:  
EYEPATTERN:QFACTOR:STATE 0;:  
TELECOMTEST:EYEPATTERN:RISE:STATE 0;:  
TELECOMTEST:EYEPATTERN:SDBASE:STATE 0;:  
TELECOMTEST:EYEPATTERN:SDTOP:  
STATE 0;:TELECOMTEST:EYEPATTERN:  
T1CROSSING:STATE 0;:TELECOMTEST:  
EYEPATTERN:T2CROSSING:STATE 0;:  
TELECOMTEST:EYEPATTERN:TLEVELS:  
MODE PERCENT;PERCENT 90,10;  
UNIT 1.000E+00,0.000E+00;:  
TELECOMTEST:EYEPATTERN:VBASE:  
STATE 0.....

### :TELecomtest:CATegory

**Function** Sets the telecom test type or queries the current setting.

**Syntax** :TELecomtest:CATegory {EYEPattern|MASK}  
:TELecomtest:CATegory?

**Example** :TELECOMTEST:CATEGORY EYEPATTERN  
:TELECOMTEST:CATEGORY? -> :TELECOMTEST:  
CATEGORY EYEPATTERN

### :TELecomtest:DISPlay

**Function** Turns ON/OFF the telecom test display or queries the current setting.

**Syntax** :TELecomtest:DISPlay {<Boolean>}  
:TELecomtest:DISPlay?

**Example** :TELECOMTEST:DISPLAY ON  
:TELECOMTEST:DISPLAY? -> :TELECOMTEST:  
DISPLAY 1

### :TELecomtest:EYEPattern?

**Function** Queries all settings related to the eye pattern.

**Syntax** :TELecomtest:EYEPattern?

**Example** :TELECOMTEST:EYEPATTERN?  
-> :TELECOMTEST:EYEPATTERN:DBERATE:  
STATE 1;:TELECOMTEST:EYEPATTERN:  
EHEIGHT:STATE 1;:TELECOMTEST:  
EYEPATTERN:EWIDTH:STATE 1;:  
TELECOMTEST:EYEPATTERN:FALL:STATE 1;:  
TELECOMTEST:EYEPATTERN:JITTER1:  
STATE 1;:TELECOMTEST:EYEPATTERN:  
JITTER2:STATE 1;:TELECOMTEST:  
EYEPATTERN:PCROSSING:STATE 1;:  
TELECOMTEST:EYEPATTERN:PDUTYCYCLE:  
STATE 1;:TELECOMTEST:EYEPATTERN:  
QFACTOR:STATE 1;:TELECOMTEST:  
EYEPATTERN:RISE:STATE 1;:TELECOMTEST:  
EYEPATTERN:SDBASE:STATE 1;:TELECOMTEST:  
EYEPATTERN:SDTOP:STATE 1;:  
TELECOMTEST:EYEPATTERN:T1CROSSING:  
STATE 1;:TELECOMTEST:EYEPATTERN:  
T2CROSSING:STATE 1;:TELECOMTEST:  
EYEPATTERN:TLEVELS:MODE PERCENT;  
PERCENT 90,10;UNIT 1.000E+00,  
0.000E+00;:TELECOMTEST:EYEPATTERN:  
VBASE:STATE 1;:TELECOMTEST:  
EYEPATTERN:VCROSSING:STATE 1;:  
TELECOMTEST:EYEPATTERN:VDARK 1;VTOP:  
STATE 1

### :TELecomtest:EYEPattern:ALL

**Function** Turns ON/OFF all eye pattern parameters.

**Syntax** :TELecomtest:EYEPattern:ALL {<Boolean>}

**Example** :TELECOMTEST:EYEPATTERN:ALL ON

### :TELecomtest:EYEPattern:<Parameter>?

**Function** Queries all settings related to the waveform parameter of the eye pattern.

**Syntax** :TELecomtest:EYEPattern:<Parameter>?  
<Parameter> = {DBERate|EHEight|EWIDth|  
FALL|JITTer<x>|PCROSSing|PDUTYcycle|  
QFACTOR|RISE|SDBase|SDTop|T1CRossing|  
T2CRossing|VBASe|VCROSSing|VTOP}  
<x> = 1 or 2

**Example** (The following is an example for DBERate.)  
:TELECOMTEST:EYEPATTERN:DBERATE?  
-> :TELECOMTEST:EYEPATTERN:DBERATE:  
STATE 1

**Description** • For the relationship between communication commands and parameters used by the DL9040/DL9140/DL9240, see appendix 4.  
• For parameter details, see the DL9040/DL9140/DL9240 User's Manual.

## 5.27 TELecomtest Group

### **:TELecomtest:EYEPattern:<Parameter>:**

#### **STATE**

**Function** Turns ON/OFF the waveform parameter of the eye pattern or queries the current setting.

**Syntax** :TELecomtest:EYEPattern:<Parameter>:  
STATE {<Boolean>}  
:TELecomtest:EYEPattern:<Parameter>:  
STATE?

<Parameter> = {DBERate|EHEight|EWIDTH|  
FALL|JITTer<x>|PCROSSing|PDUTYcycle|  
QFACTOR|RISE|SDBase|SDTop|T1CRossing|  
T2CRossing|VBASe|VCRossing|VTOP}  
<x> = 1 or 2

**Example** (The following is an example for DBERate.)  
:TELECOMTEST:EYEPATTERN:DBERATE:  
STATE ON  
:TELECOMTEST:EYEPATTERN:DBERATE:STATE?  
-> :TELECOMTEST:EYEPATTERN:DBERATE:  
STATE 1

### **:TELecomtest:EYEPattern:<Parameter>:**

#### **VALue?**

**Function** Queries the waveform parameter value of the eye pattern.

**Syntax** :TELecomtest:EYEPattern:<Parameter>:  
VALue?

<Parameter> = {DBERate|EHEight|EWIDTH|  
FALL|JITTer<x>|PCROSSing|PDUTYcycle|  
QFACTOR|RISE|SDBase|SDTop|T1CRossing|  
T2CRossing|VBASe|VCRossing|VTOP}  
<x> = 1 or 2

**Example** (The following is an example for DBERate.)  
:TELECOMTEST:EYEPATTERN:DBERATE:VALUE?  
-> :TELECOMTEST:EYEPATTERN:DBERATE:  
VALUE 1.000E+00

### **:TELecomtest:EYEPattern:TLEVELs?**

**Function** Queries all settings related to the threshold level of the eye pattern.

**Syntax** :TELecomtest:EYEPattern:TLEVELs?

**Example** :TELECOMTEST:EYEPATTERN:TLEVELS?  
-> :TELECOMTEST:EYEPATTERN:TLEVELS:  
MODE PERCENT;PERCENT 90,10;  
UNIT 1.000E+00,0.000E+00

### **:TELecomtest:EYEPattern:TLEVELs:MODE**

**Function** Sets the unit of the threshold level of the eye pattern or queries the current setting.

**Syntax** :TELecomtest:EYEPattern:TLEVELs:  
MODE {PERCent|UNIT}  
:TELecomtest:EYEPattern:TLEVELs:MODE?

**Example** :TELECOMTEST:EYEPATTERN:TLEVELS:  
MODE PERCENT  
:TELECOMTEST:EYEPATTERN:TLEVELS:MODE?  
-> :TELECOMTEST:EYEPATTERN:TLEVELS:  
MODE PERCENT

### **:TELecomtest:EYEPattern:TLEVELs:**

#### **PERCent**

**Function** Sets the threshold level of the eye pattern as a percentage or queries the current setting.

**Syntax** :TELecomtest:EYEPattern:TLEVELs:  
PERCent {<NRf>,<NRf>}  
:TELecomtest:EYEPattern:TLEVELs:  
PERCent?

<NRf> = 0 to 100 (%)

**Example** :TELECOMTEST:EYEPATTERN:TLEVELS:  
PERCENT 90,10  
:TELECOMTEST:EYEPATTERN:TLEVELS:  
PERCENT? -> :TELECOMTEST:EYEPATTERN:  
TLEVELS:PERCENT 90,10

### **:TELecomtest:EYEPattern:TLEVELs:UNIT**

**Function** Sets the threshold level of the eye pattern in UNIT or queries the current setting.

**Syntax** :TELecomtest:EYEPattern:TLEVELs:  
UNIT {<NRf>,<NRf>|<Voltage>,<Voltage>|  
<Current>,<Current>}  
:TELecomtest:EYEPattern:TLEVELs:UNIT?  
<NRf>,<Voltage>, and <Current> = See the DL9000  
User's Manual.

**Example** :TELECOMTEST:EYEPATTERN:TLEVELS:  
UNIT 1,0  
:TELECOMTEST:EYEPATTERN:TLEVELS:UNIT?  
-> :TELECOMTEST:EYEPATTERN:TLEVELS:  
UNIT 1.000E+00,0.000E+00

### **:TELecomtest:EYEPattern:VDARK**

**Function** Sets the dark level (zero light level) or queries the current setting.

**Syntax** :TELecomtest:EYEPattern:VDARK {<NRf>|  
<Voltage>|<Current>}  
:TELecomtest:EYEPattern:VDARK?  
<NRf>,<Voltage>, and <Current> = See the DL9000  
User's Manual.

**Example** :TELECOMTEST:EYEPATTERN:VDARK 1.000E+00  
:TELECOMTEST:EYEPATTERN:VDARK?  
-> :TELECOMTEST:EYEPATTERN:  
VDARK 1.000E+00

**:TELEcomtest:MASK?**

Function Queries all settings related to the mask test.

Syntax :TELEcomtest:MASK?

Example :TELECOMTEST:MASK? -> :TELECOMTEST:  
MASK:ELEMENT1:PSPCOUNT:STATE 1;;  
TELECOMTEST:MASK:ELEMENT1:PWCOUNT:  
STATE 1;;TELECOMTEST:MASK:ELEMENT1:  
SPCOUNT:STATE 1;;TELECOMTEST:MASK:  
ELEMENT1:WCOUNT:STATE 1;;TELECOMTEST:  
MASK:ELEMENT2:PSPCOUNT:STATE 1;;  
TELECOMTEST:MASK:ELEMENT2:PWCOUNT:  
STATE 1;;TELECOMTEST:MASK:ELEMENT2:  
SPCOUNT:STATE 1;;TELECOMTEST:MASK:  
ELEMENT2:WCOUNT:STATE 1;;TELECOMTEST:  
MASK:ELEMENT3:PSPCOUNT:STATE 1;;  
TELECOMTEST:MASK:ELEMENT3:PWCOUNT:  
STATE 1;;TELECOMTEST:MASK:ELEMENT3:  
SPCOUNT:STATE 1;;TELECOMTEST:MASK:  
ELEMENT3:WCOUNT:STATE 1;;TELECOMTEST:  
MASK:ELEMENT4:PSPCOUNT:STATE 1;;  
TELECOMTEST:MASK:ELEMENT4:PWCOUNT:  
STATE 1;;TELECOMTEST:MASK:ELEMENT4:  
SPCOUNT:STATE 1;;TELECOMTEST:MASK:  
ELEMENT4:WCOUNT:STATE 1

**:TELEcomtest:MASK:ELEMENT<x>?**

Function Queries all settings related to the element used in the mask test.

Syntax :TELEcomtest:MASK:ELEMENT<x>?  
<x> = 1 to 4

Example :TELECOMTEST:MASK:ELEMENT1?  
-> :TELECOMTEST:MASK:ELEMENT1:PSPCOUNT:  
STATE 1;;TELECOMTEST:MASK:ELEMENT1:  
PWCOUNT:STATE 1;;TELECOMTEST:MASK:  
ELEMENT1:SPCOUNT:STATE 1;;TELECOMTEST:  
MASK:ELEMENT1:WCOUNT:STATE 1

**:TELEcomtest:MASK:ELEMENT<x>:ALL**

Function Turns ON/OFF all items of the element.

Syntax :TELEcomtest:MASK:ELEMENT<x>:ALL  
{<Boolean>}  
<x> = 1 to 4

Example :TELECOMTEST:MASK:ELEMENT1:ALL ON

**:TELEcomtest:MASK:ELEMENT<x>:****PSPCount? (Sample Point Count %)**

Function Queries the settings related to the error rate for the number of sampled data points of the element.

Syntax :TELEcomtest:MASK:ELEMENT<x>:PSPCount?  
<x> = 1 to 4

Example :TELECOMTEST:MASK:ELEMENT1:PSPCOUNT?  
-> :TELECOMTEST:MASK:ELEMENT1:PSPCOUNT:  
STATE 1

**:TELEcomtest:MASK:ELEMENT<x>:****PSPCount:STATE**

Function Turns ON/OFF the measurement of the error rate for the number of sampled data points of the element or queries the current setting.

Syntax :TELEcomtest:MASK:ELEMENT<x>:PSPCount:  
STATE {<Boolean>}  
:TELEcomtest:MASK:ELEMENT<x>:PSPCount:  
STATE?  
<x> = 1 to 4

Example :TELECOMTEST:MASK:ELEMENT1:PSPCOUNT:  
STATE ON  
:TELECOMTEST:MASK:ELEMENT1:PSPCOUNT:  
STATE? -> :TELECOMTEST:MASK:ELEMENT1:  
PSPCOUNT:STATE 1

**:TELEcomtest:MASK:ELEMENT<x>:****PSPCount:VALUE?**

Function Queries the error rate for the number of sampled data points of the element.

Syntax :TELEcomtest:MASK:ELEMENT<x>:PSPCount:  
VALUE?  
<x> = 1 to 4

Example :TELECOMTEST:MASK:ELEMENT1:PSPCOUNT:  
VALUE? -> :TELECOMTEST:MASK:ELEMENT1:  
PSPCOUNT:VALUE 1.000E+00

**:TELEcomtest:MASK:ELEMENT<x>:****PWCount? (Wave Count %)**

Function Queries the settings related to the error rate for the acquisition count of the element.

Syntax :TELEcomtest:MASK:ELEMENT<x>:PWCount?  
<x> = 1 to 4

Example :TELECOMTEST:MASK:ELEMENT1:PWCOUNT?  
-> :TELECOMTEST:MASK:ELEMENT1:PWCOUNT:  
STATE 1

## 5.27 TELecomtest Group

### **:TELecomtest:MASK:ELEMent<x>:PWCount:STATE**

**Function** Turns ON/OFF the measurement of the error rate for the acquisition count of the element or queries the current setting.

**Syntax** :TELecomtest:MASK:ELEMent<x>:PWCount:STATE {<Boolean>}  
:TELecomtest:MASK:ELEMent<x>:PWCount:STATE?  
<x> = 1 to 4

**Example** :TELECOMTEST:MASK:ELEMENT1:PWCOUNT:STATE ON  
:TELECOMTEST:MASK:ELEMENT1:PWCOUNT:STATE? -> :TELECOMTEST:MASK:ELEMENT1:PWCOUNT:STATE 1

### **:TELecomtest:MASK:ELEMent<x>:PWCount:VALue?**

**Function** Queries the error rate for the acquisition count of the element.

**Syntax** :TELecomtest:MASK:ELEMent<x>:PWCount:VALue?  
<x> = 1 to 4

**Example** :TELECOMTEST:MASK:ELEMENT1:PWCOUNT:VALUE? -> :TELECOMTEST:MASK:ELEMENT1:PWCOUNT:VALUE 1.000E+00

### **:TELecomtest:MASK:ELEMent<x>:SPCount? (Sample Point Count)**

**Function** Queries the settings related to the number of sampled data points for the element that results in error.

**Syntax** :TELecomtest:MASK:ELEMent<x>:SPCount?<x> = 1 to 4

**Example** :TELECOMTEST:MASK:ELEMENT1:SPCOUNT? -> :TELECOMTEST:MASK:ELEMENT1:SPCOUNT:STATE 1

### **:TELecomtest:MASK:ELEMent<x>:SPCount:STATE**

**Function** Turns ON/OFF the measurement of the number of sampled data points for the element that results in error or queries the current setting.

**Syntax** :TELecomtest:MASK:ELEMent<x>:SPCount:STATE {<Boolean>}  
:TELecomtest:MASK:ELEMent<x>:SPCount:STATE?  
<x> = 1 to 4

**Example** :TELECOMTEST:MASK:ELEMENT1:SPCOUNT:STATE ON  
:TELECOMTEST:MASK:ELEMENT1:SPCOUNT:STATE? -> :TELECOMTEST:MASK:ELEMENT1:SPCOUNT:STATE 1

### **:TELecomtest:MASK:ELEMent<x>:SPCount:VALue?**

**Function** Queries the number of sampled data points for the element that resulted in error.

**Syntax** :TELecomtest:MASK:ELEMent<x>:SPCount:VALue?  
<x> = 1 to 4

**Example** :TELECOMTEST:MASK:ELEMENT1:SPCOUNT:VALUE? -> :TELECOMTEST:MASK:ELEMENT1:SPCOUNT:VALUE 1

### **:TELecomtest:MASK:ELEMent<x>:WCOunt? (Wave Count)**

**Function** Queries the settings related to the acquisition count for the element that results in error.

**Syntax** :TELecomtest:MASK:ELEMent<x>:WCOunt?<x> = 1 to 4

**Example** :TELECOMTEST:MASK:ELEMENT1:WCOUNT? -> :TELECOMTEST:MASK:ELEMENT1:WCOUNT:STATE 1

### **:TELecomtest:MASK:ELEMent<x>:WCOunt:STATE**

**Function** Turns ON/OFF the measurement of the acquisition count for the element that results in error or queries the current setting.

**Syntax** :TELecomtest:MASK:ELEMent<x>:WCOunt:STATE {<Boolean>}  
:TELecomtest:MASK:ELEMent<x>:WCOunt:STATE?  
<x> = 1 to 4

**Example** :TELECOMTEST:MASK:ELEMENT1:WCOUNT:STATE ON  
:TELECOMTEST:MASK:ELEMENT1:WCOUNT:STATE? -> :TELECOMTEST:MASK:ELEMENT1:WCOUNT:STATE 1

### **:TELecomtest:MASK:ELEMent<x>:WCOunt:VALue?**

**Function** Queries the acquisition count for the element that resulted in error.

**Syntax** :TELecomtest:MASK:ELEMent<x>:WCOunt:VALue?  
<x> = 1 to 4

**Example** :TELECOMTEST:MASK:ELEMENT1:WCOUNT:VALUE? -> :TELECOMTEST:MASK:ELEMENT1:WCOUNT:VALUE 1

**:TELecomtest:MMODE**

Function Turns ON/OFF the computed waveform or queries the current setting.

Syntax :TELecomtest:MMODE {<Boolean>}  
:TELecomtest:MMODE?

Example :TELECOMTEST:MMODE ON  
:TELECOMTEST:MMODE? -> :TELECOMTEST:  
MMODE 1

**:TELecomtest:TRACe**

Function Sets the source trace of the telecom test or queries the current setting.

Syntax :TELecomtest:TRACe {<NRf>}  
:TELecomtest:TRACe?  
<NRf> = 1 to 8

Example :TELECOMTEST:TRACE 1  
:TELECOMTEST:TRACE? -> :TELECOMTEST:  
TRACE 1

**:TELecomtest:TRANge (Time Range)**

Function Sets the measurement range of the telecom test or queries the current setting.

Syntax :TELecomtest:TRANge {<NRf>,<NRf>}  
:TELecomtest:TRANge?  
<NRf> = -5 to 5 (div)

Example :TELECOMTEST:TRANGE -5,0  
:TELECOMTEST:TRANGE? -> :TELECOMTEST:  
TRANGE 0.000E+00,-5.000E+00

**:TELecomtest:WINDow**

Function Sets the measurement source window of the telecom test or queries the current setting.

Syntax :TELecomtest:WINDow {MAIN|Z1|Z2}  
:TELecomtest:WINDow?

Example :TELECOMTEST:WINDOW MAIN  
:TELECOMTEST:WINDOW? -> :TELECOMTEST:  
WINDOW MAIN

---

## 5.28 TIMEbase Group

### **:TIMEbase?**

Function Queries all settings related to the time base.

Syntax :TIMEbase?

Example :TIMEBASE? -> :TIMEBASE:TDIV 1.000E-06

### **:TIMEbase:SRATE? (Sample RATE)**

Function Queries the sample rate or queries the current setting.

Syntax :TIMEbase:SRATE?

Example :TIMEBASE:SRATE? -> :TIMEBASE:  
SRATE 125.0E+06

### **:TIMEbase:TDIV**

Function Sets the T/div value or queries the current setting.

Syntax :TIMEbase:TDIV {<Time>}

:TIMEbase:TDIV?

<Time> = 500 ps to 50 s

Example :TIMEBASE:TDIV 1NS  
:TIMEBASE:TDIV? -> :TIMEBASE:  
TDIV 1.000E-06

## 5.29 TRIGger Group

### **:TRIGger?**

**Function** Queries all settings related to the trigger.

**Syntax** :TRIGger?

**Example** :TRIGGER? -> :TRIGGER:ACTION:  
ACQCOUNT 1;BUZZER 0;HCOPY 1;MAIL:  
INTERVAL OFF;MODE 0;:TRIGGER:ACTION:  
MODE ACONDITION;SAVE 1;:TRIGGER:  
TYPE EICYCLE;CLOCK:SOURCE 1;  
POLARITY RISE;:TRIGGER:DELAY:  
EDGECOUNT:COUNT 1;:TRIGGER:DELAY:  
MODE 1;POLARITY FALL;SOURCE 4;  
TIME 1.000E+00;TYPE EDGECOUNT;:  
TRIGGER:EINTERVAL:EVENT1:TYPE EDGE;  
CLOCK:SOURCE 1;POLARITY FALL;:  
TRIGGER:EINTERVAL:EVENT1:ESTATE:  
SOURCE 1;POLARITY FALL;:TRIGGER:  
EINTERVAL:EVENT1:CANBUS:  
ACK DONTCARE;BRATE 1000000;DATA:  
BORDER BIG;CONDITION DONTCARE;  
DATA1 0.0000000E+00;  
DATA2 255.000000E+00;DLC 8;MSBSLB 7,0;  
PATTERN "111001010110010001111000100  
1001100101010001000010001111111110  
10";SIGN UNSIGN;:TRIGGER:EINTERVAL:  
EVENT1:CANBUS:IDEXT:PATTERN "XXXXXX  
XXXXXXXXXXXXXXXXXXXXXXXX";:TRIGGER:  
EINTERVAL:EVENT1:CANBUS:IDOR:ID1:  
ACK DONTCARE;DATA:BORDER BIG;  
CONDITION DONTCARE;  
DATA1 0.0000000E+00;  
DATA2 255.000000E+00;DLC 8;  
MSBSLB 7,0;PATTERN "0000000100100011  
01000101011001111000100110101011100  
11011101111";SIGN UNSIGN;:TRIGGER:  
EINTERVAL:EVENT1:CANBUS:IDOR:ID1:  
FORMAT STD;IDEXT:PATTERN "1101010  
1111001101111011110000";:TRIGGER:  
EINTERVAL:EVENT1:CANBUS:IDOR:ID1:  
IDSTD:PATTERN "00100100011";:TRIGGER:  
EINTERVAL:EVENT1:CANBUS:IDOR:ID1:  
MODE 0;RTR DATA;:TRIGGER:EINTERVAL:  
EVENT1:CANBUS:IDOR:ID2:ACK DONTCARE;  
DATA:BORDER BIG;CONDITION DONTCARE;  
DATA1 0.0000000E+00;  
DATA2 255.000000E+00;DLC 8;  
MSBSLB 7,0;PATTERN "111111011011100  
101110101001100001110110010101000011  
001000010000";SIGN UNSIGN;:TRIGGER:  
EINTERVAL:EVENT1:CANBUS:IDOR:ID2:  
FORMAT STD;IDEXT:PATTERN "1001000  
1101000101011001111000";:TRIGGER:  
EINTERVAL:EVENT1:CANBUS:IDOR:ID2:  
IDSTD:PATTERN "10001010110";:TRIGGER:  
EINTERVAL:EVENT1:CANBUS:IDOR:ID2:  
MODE 0.....

### **:TRIGger:ACTion?**

**Function** Queries all settings related to action-on-trigger.

**Syntax** :TRIGger:ACTion?

**Example** :TRIGGER:ACTION? -> :TRIGGER:ACTION:  
ACQCOUNT 1;BUZZER 0;  
HCOPY 1;MAIL:INTERVAL OFF;MODE 0;  
:TRIGGER:ACTION:MODE ACONDITION;  
SAVE 1

### **:TRIGger:ACTion:ACQCount**

**Function** Sets the action count of action-on-trigger or queries the current setting.

**Syntax** :TRIGger:ACTion:ACQCount {<NRf>|  
INFinite}  
:TRIGger:ACTion:ACQCount?  
<NRf> = 1 to 1000000

**Example** :TRIGGER:ACTION:ACQCOUNT 10  
:TRIGGER:ACTION:ACQCOUNT? -> :TRIGGER:  
ACTION:ACQCOUNT 10

### **:TRIGger:ACTion:BUZZer**

**Function** Sets whether to sound a buzzer when an action is activated or queries the current setting.

**Syntax** :TRIGger:ACTion:BUZZer {<Boolean>}  
:TRIGger:ACTion:BUZZer?

**Example** :TRIGGER:ACTION:BUZZER ON  
:TRIGGER:ACTION:BUZZER? -> :TRIGGER:  
ACTION:BUZZER 1

### **:TRIGger:ACTion:HCOPY**

**Function** Sets whether or not to output screen image data (ON/OFF) when an action is activated, or queries the current setting.

**Syntax** :TRIGger:ACTion:HCOPY {<Boolean>}  
:TRIGger:ACTion:HCOPY?

**Example** :TRIGGER:ACTION:HCOPY ON  
:TRIGGER:ACTION:HCOPY? -> :TRIGGER:  
ACTION:HCOPY 1

### **:TRIGger:ACTion:MAIL?**

**Function** Queries all settings related to the mail transmission when an action is activated.

**Syntax** :TRIGger:ACTion:MAIL?

**Example** :TRIGGER:ACTION:MAIL? -> :TRIGGER:  
ACTION:MAIL:INTERVAL 10;MODE 1

## 5.29 TRIGger Group

### **:TRIGger:ACTion:MAIL:INTERval**

Function Sets the interval at which to send mail when an action is activated or queries the current setting.

Syntax :TRIGger:ACTion:MAIL:INTERval  
{OFF|<Nrf>}  
:TRIGger:ACTion:MAIL:INTERval?  
<Nrf> = 1 to 1440 (min)

Example :TRIGGER:ACTION:MAIL:INTERVAL 10  
:TRIGGER:ACTION:MAIL:INTERVAL?  
-> :TRIGGER:ACTION:MAIL:INTERVAL 10

### **:TRIGger:ACTion:MAIL:MODE**

Function Sets whether to send mail when an action is activated or queries the current setting.

Syntax :TRIGger:ACTion:MAIL:MODE {<Boolean>}  
:TRIGger:ACTion:MAIL:MODE?

Example :TRIGGER:ACTION:MAIL:MODE ON:TRIGGER:  
ACTION:MAIL:MODE? -> :TRIGGER:ACTION:  
MAIL:MODE 1

### **:TRIGger:ACTion:MODE**

Function Sets the action-on-trigger mode or queries the current setting.

Syntax :TRIGger:ACTion:MODE {ACONdition|OFF}  
:TRIGger:ACTion:MODE?

Example :TRIGGER:ACTION:MODE ACONDITION  
:TRIGGER:ACTION:MODE? -> :TRIGGER:  
ACTION:MODE ACONDITION

### **:TRIGger:ACTion:SAVE**

Function Sets whether to save the waveform data to the storage medium (ON/OFF) when an action is activated or queries the current setting.

Syntax :TRIGger:ACTion:SAVE {<Boolean>}  
:TRIGger:ACTion:SAVE?

Example :TRIGGER:ACTION:SAVE ON  
:TRIGGER:ACTION:SAVE? -> :TRIGGER:  
ACTION:SAVE 1

### **:TRIGger:ACTion:START**

Function Starts the action-on-trigger.

Syntax :TRIGger:ACTion:START

Example :TRIGGER:ACTION:START

### **:TRIGger:ACTion:STOP**

Function Stops the action-on-trigger.

Syntax :TRIGger:ACTion:STOP

Example :TRIGGER:ACTION:STOP

### **:TRIGger:CLOCK?**

Function Queries all settings related to the clock channel.

Syntax :TRIGger:CLOCK?

Example :TRIGGER:CLOCK? -> :TRIGGER:CLOCK:  
SOURCE 1;POLARITY RISE

### **:TRIGger:CLOCK:POLarity**

Function Sets the polarity of the clock channel or queries the current setting.

Syntax :TRIGger:CLOCK:POLarity {ENTER|EXIT|  
FALL|RISE}  
:TRIGger:CLOCK:POLarity?

Example :TRIGGER:CLOCK:POLARITY RISE  
:TRIGGER:CLOCK:POLARITY? -> :TRIGGER:  
CLOCK:POLARITY RISE

Description • This command is invalid when :TRIGger:CLOCK:  
SOURCE NONE.

- {ENTER|EXIT} is valid when :TRIGger:SOURCE:  
CHANNEL<x>:WINDOW ON. For all other cases,  
{FALL|RISE} is valid.

- This command is valid when :TRIGger:TYPE  
PSTATE|STATE.

### **:TRIGger:CLOCK:SOURCE**

Function Sets the source waveform of the clock channel or queries the current setting.

Syntax :TRIGger:CLOCK:SOURCE {<Nrf>|NONE}  
:TRIGger:CLOCK:SOURCE?  
<Nrf> = 1 to 4

Example :TRIGGER:CLOCK:SOURCE NONE  
:TRIGGER:CLOCK:SOURCE? -> :TRIGGER:  
CLOCK:SOURCE NONE

Description This command is valid when :TRIGger:TYPE  
PSTATE|STATE.

### **:TRIGger:DELAY?**

Function Queries all settings related to the trigger delay.

Syntax :TRIGger:DELAY?

Example :TRIGGER:DELAY? -> :TRIGGER:DELAY:  
EDGECOUNT:COUNT 1;;  
TRIGGER:DELAY:MODE 1;POLARITY FALL;  
SOURCE 4;TIME 1.000E+00;TYPE EDGECOUNT

### **:TRIGger:DELAY:EDGEcount?**

Function Queries all settings related to edge count of the trigger delay.

Syntax :TRIGger:DELAY:EDGEcount?

Example :TRIGGER:DELAY:EDGECOUNT? -> :TRIGGER:  
DELAY:EDGECOUNT:COUNT 1

### **:TRIGger:DELAY:EDGEcount:COUNT**

Function Sets the edge count value of the trigger delay or queries the current setting.

Syntax :TRIGger:DELAY:EDGEcount:COUNT {<Nrf>}  
:TRIGger:DELAY:EDGEcount:COUNT?  
<Nrf> = 1 to 100000000

Example :TRIGGER:DELAY:EDGECOUNT:COUNT 1  
:TRIGGER:DELAY:EDGECOUNT:COUNT?  
-> :TRIGGER:DELAY:EDGECOUNT:COUNT 1



**:TRIGger:DElay:MODE**

**Function** Turns ON/OFF the trigger delay or queries the current setting.

**Syntax** :TRIGger:DElay:MODE {<Boolean>}  
:TRIGger:DElay:MODE?

**Example** :TRIGGER:DELAY:MODE ON  
:TRIGGER:DELAY:MODE? -> :TRIGGER:DELAY:  
MODE 1

**:TRIGger:DElay:POLarity**

**Function** Sets the edge polarity the trigger delay or queries the current setting.

**Syntax** :TRIGger:DElay:POLarity {FALL|RISE}  
:TRIGger:DElay:POLarity?

**Example** :TRIGGER:DELAY:POLARITY RISE  
:TRIGGER:DELAY:POLARITY? -> :TRIGGER:  
DELAY:POLARITY RISE

**Description** This command is valid when :TRIGger:DElay:  
TYPE EDGecount|FEADelay.

**:TRIGger:DElay:SOURce**

**Function** Sets the edge source the trigger delay or queries the current setting.

**Syntax** :TRIGger:DElay:SOURce {<NRf>|EXTernal}  
:TRIGger:DElay:SOURce?  
<NRf> = 1 to 4

**Example** :TRIGGER:DELAY:SOURCE 1  
:TRIGGER:DELAY:SOURCE? -> :TRIGGER:  
DELAY:SOURCE 1

**Description** This command is valid when :TRIGger:DElay:  
TYPE EDGecount|FEADelay.

**:TRIGger:DElay:TIME**

**Function** Sets the delay value the trigger delay or queries the current setting.

**Syntax** :TRIGger:DElay:TIME {<Time>}  
:TRIGger:DElay:TIME?  
<Time> = 0 s to 10 s in 5-ps steps

**Example** :TRIGGER:DELAY:TIME 1S  
:TRIGGER:DELAY:TIME? -> :TRIGGER:DELAY:  
TIME 1.000E+00

**Description** This command is valid when :TRIGger:DElay:  
TYPE BYTime|FEADelay.

**:TRIGger:DElay:TYPE**

**Function** Sets the trigger delay type or queries the current setting.

**Syntax** :TRIGger:DElay:TYPE {BYTime|EDGecount|  
FEADelay}  
:TRIGger:DElay:TYPE?

**Example** :TRIGGER:DELAY:TYPE BYTIME  
:TRIGGER:DELAY:TYPE? -> :TRIGGER:DELAY:  
TYPE BYTIME

**:TRIGger:EINterval?**

**Function** Queries all settings related to the event interval.

**Syntax** :TRIGger:EINterval?

**Example** :TRIGGER:EINTERVAL? -> :TRIGGER:  
EINTERVAL:EVENT1:TYPE EDGE;CLOCK:  
SOURCE 1;POLARITY FALL;;TRIGGER:  
EINTERVAL:EVENT1:ESTATE:SOURCE 1;  
POLARITY FALL;;TRIGGER:EINTERVAL:  
EVENT1:I2CBUS:ADATA:BIT10ADDRESS:  
PATTERN "10111011111";:TRIGGER:  
EINTERVAL:EVENT1:I2CBUS:ADATA:  
BIT7ADDRESS:PATTERN "11011110";:  
TRIGGER:EINTERVAL:EVENT1:I2CBUS:ADATA:  
BIT7APSUB:ADDRESS:PATTERN "10101011";:  
TRIGGER:EINTERVAL:EVENT1:I2CBUS:ADATA:  
BIT7APSUB:SADDRESS:PATTERN "11001101";:  
TRIGGER:EINTERVAL:EVENT1:I2CBUS:ADATA:  
TYPE BIT10ADDRESS;;TRIGGER:EINTERVAL:  
EVENT1:I2CBUS:CLOCK:SOURCE 1;;TRIGGER:  
EINTERVAL:EVENT1:I2CBUS:DATA:BYTE 1;  
CONDITION TRUE;DPOSITION 1;MODE 1;  
PATTERN1 "10101011";  
PATTERN2 "10101010";  
PATTERN3 "10101111";  
PATTERN4 "10101011";PMODE DONTCARE;  
SOURCE 1;;TRIGGER:EINTERVAL:EVENT1:  
I2CBUS:GCALL:BIT7MADDRESS:  
PATTERN "1010101";:TRIGGER:EINTERVAL:  
EVENT1:I2CBUS:GCALL:  
SBYTE BIT7MADDRESS;;TRIGGER:EINTERVAL:  
EVENT1:I2CBUS:MODE ADATA;NAIGNORE:  
HSMODE 1;RACCESS 1;SBYTE 1;;TRIGGER:  
EINTERVAL:EVENT1:I2CBUS:SBHSMODE:  
TYPE HSMODE;;TRIGGER:EINTERVAL:EVENT1:  
SPATTERN:BITRATE 1.000E+00;CLOCK:  
MODE 1;POLARITY FALL;SOURCE 1;;  
TRIGGER:EINTERVAL:EVENT1:SPATTERN:CS 1;  
DATA:ACTIVE HIGH;SOURCE 1;;TRIGGER:  
EINTERVAL:EVENT1:SPATTERN:LATCH:  
SOURCE 1;POLARITY FALL;;TRIGGER:  
EINTERVAL:EVENT1:SPATTERN:  
PATTERN "1100110111101111".....

## 5.29 TRIGger Group

### :TRIGger:EINterval:EVENT<x>?

Function Queries all settings related to the event.

Syntax :TRIGger:EINterval:EVENT<x>?

<x> = 1 or 2

Example :TRIGGER:EINTERVAL:EVENT1? -> :TRIGGER:  
EINTERVAL:EVENT1:TYPE EDGE;  
CLOCK:SOURCE 1;POLARITY FALL;:TRIGGER:  
EINTERVAL:EVENT1:ESTATE:SOURCE 1;  
POLARITY FALL;:TRIGGER:EINTERVAL:  
EVENT1:I2CBUS:ADATA:BIT10ADDRESS:  
PATTERN "10111011111";:TRIGGER:E  
INTERVAL:EVENT1:I2CBUS:ADATA:  
BIT7ADDRESS:PATTERN "11011110";:  
TRIGGER:EINTERVAL:EVENT1:I2CBUS:ADATA:  
BIT7APSUB:ADDRESS:PATTERN "10101011";:  
TRIGGER:EINTERVAL:EVENT1:I2CBUS:ADATA:  
BIT7APSUB:SADDRESS:PATTERN "11001101";:  
TRIGGER:EINTERVAL:EVENT1:I2CBUS:ADATA:  
TYPE BIT10ADDRESS;:TRIGGER:EINTERVAL:  
EVENT1:I2CBUS:CLOCK:SOURCE 1;:TRIGGER:  
EINTERVAL:EVENT1:I2CBUS:DATA:BYTE 1;  
CONDITION TRUE;DPOSITION 1;MODE 1;  
PATTERN1 "10101011";  
PATTERN2 "10101010";  
PATTERN3 "10101111";  
PATTERN4 "10101011";PMODE DONTCARE;  
SOURCE 1;:TRIGGER:EINTERVAL:EVENT1:  
I2CBUS:GCALL:BIT7MADDRESS:  
PATTERN "1010101";:TRIGGER:EINTERVAL:  
EVENT1:I2CBUS:GCALL:  
SBYTE BIT7MADDRESS;:TRIGGER:EINTERVAL:  
EVENT1:I2CBUS:MODE ADATA;NAIGNORE:  
HSMODE 1;RACCESS 1;SBYTE 1;:TRIGGER:  
EINTERVAL:EVENT1:I2CBUS:SBHSMODE:  
TYPE HSMODE;:TRIGGER:EINTERVAL:EVENT1:  
SPATTERN:BITRATE 1.000E+00;CLOCK:  
MODE 1;POLARITY FALL;SOURCE 1;:  
TRIGGER:EINTERVAL:EVENT1:SPATTERN:CS 1;  
DATA:ACTIVE HIGH;SOURCE 1;:TRIGGER:  
EINTERVAL:EVENT1:SPATTERN:LATCH:  
SOURCE 1;POLARITY FALL;:TRIGGER:  
EINTERVAL:EVENT1:SPATTERN:  
PATTERN "1100110111101111".....

Description EVENT2 is valid when :TRIGger:TYPE  
EIDelay|EISequence.

### :TRIGger:EINterval:EVENT<x>:CANBus?

Function Queries all settings related to the CAN bus signal trigger of the event.

Syntax :TRIGger:EINterval:EVENT<x>:CANBus?

<x> = 1 or 2

Example :TRIGGER:EINTERVAL:EVENT1:CANBUS?  
-> :TRIGGER:EINTERVAL:EVENT1:CANBUS:  
ACK DONTCARE;BRATE 1000000;DATA:  
BORDER BIG;CONDITION DONTCARE;  
DATA1 0.0000000E+00;DATA2 255.000000E+00;  
DLC 8;MSBLSB 7,0;PATTERN "111001010  
11001000111100010010011001010100010  
0001000111111111010";SIGN UNSIGN;:  
TRIGGER:EINTERVAL:EVENT1:CANBUS:  
IDEXT:PATTERN "XXXXXXXXXXXXXXXXXXXXX  
XXXXXXXX";:TRIGGER:EINTERVAL:EVENT1:  
CANBUS:IDOR:ID1:ACK DONTCARE;DATA:  
BORDER BIG;CONDITION DONTCARE;  
DATA1 0.0000000E+00;  
DATA2 255.000000E+00;DLC 8;MSBLSB 7,0;  
PATTERN "000000010010001101000101011  
00111100010011010101111001101111011  
11";SIGN UNSIGN;:TRIGGER:EINTERVAL:  
EVENT1:CANBUS:IDOR:ID1:FORMAT STD;  
IDEXT:PATTERN ?110101011110011011110  
11110000?;:TRIGGER:EINTERVAL:EVENT1:  
CANBUS:IDOR:ID1:IDSTD:  
PATTERN ?00100100011?;:TRIGGER:  
EINTERVAL:EVENT1:CANBUS:IDOR:ID1:  
MODE 0;RTR DATA;:TRIGGER:EINTERVAL:  
EVENT1:CANBUS:IDOR:ID2:ACK DONTCARE;  
DATA:BORDER BIG;CONDITION DONTCARE;  
DATA1 0.0000000E+00;  
DATA2 255.000000E+00;DLC 8;MSBLSB 7,0;  
PATTERN "111111101101110010111010100  
11000011101100101010000110010000100  
00";SIGN UNSIGN;:TRIGGER:EINTERVAL:  
EVENT1:CANBUS:IDOR:ID2:FORMAT STD;  
IDEXT:PATTERN "100100011010001010110  
01111000";:TRIGGER:EINTERVAL:EVENT1:  
CANBUS:IDOR:ID2:IDSTD:  
PATTERN ?10001010110?;:TRIGGER:  
EINTERVAL:EVENT1:CANBUS:IDOR:ID2:  
MODE 0.....

**:TRIGger:EINterval:EVENT<x>:CANBus:ACK**

**Function** Sets the ACK condition of the CAN bus signal trigger or queries the current setting.

**Syntax** :TRIGger:EINterval:EVENT<x>:CANBus:  
ACK {ACK|ACKBoth|DONTcare|NONack}  
:TRIGger:EINterval:EVENT<x>:CANBus:  
ACK?  
<x> = 1 or 2

**Example** :TRIGGER:EINTERVAL:EVENT1:CANBUS:  
ACK ACK  
:TRIGGER:EINTERVAL:EVENT1:CANBUS:  
ACK? -> :TRIGGER:EINTERVAL:EVENT1:  
CANBUS:ACK ACK

**:TRIGger:EINterval:EVENT<x>:CANBus:BRATe**

**Function** Sets the bit rate (data transfer rate) of the CAN bus signal trigger or queries the current setting.

**Syntax** :TRIGger:EINterval:EVENT<x>:CANBus:  
BRATe {<Nrf>|USER,<Nrf>}  
:TRIGger:EINterval:EVENT<x>:CANBus:  
BRATe?  
<x> = 1 or 2  
<Nrf> = 33300, 83300, 125000, 250000, 500000,  
1000000  
<Nrf> of USER = See the User's Manual (IM701310-  
01E).

**Example** :TRIGGER:EINTERVAL:EVENT1:CANBUS:  
BRATE 83300  
:TRIGGER:EINTERVAL:EVENT1:CANBUS:  
BRATE? -> :TRIGGER:EINTERVAL:EVENT1:  
CANBUS:BRATE 83300

**:TRIGger:EINterval:EVENT<x>:CANBus:DATA?**

**Function** Queries all settings related to the CAN bus signal trigger data.

**Syntax** :TRIGger:EINterval:EVENT<x>:CANBus:  
DATA?  
<x> = 1 or 2

**Example** :TRIGGER:EINTERVAL:EVENT1:CANBUS:  
DATA? ->:TRIGGER:EINTERVAL:EVENT1:  
CANBUS:DATA:BORDER BIG;  
CONDITION DONTCARE;  
DATA1 0.0000000E+00;  
DATA2 255.00000E+00;DLC 8;MSLSB 7,0;  
PATTERN "1110010101100100011110001001  
00110010101000100001000111111111010"  
;SIGN UNSIGN

**:TRIGger:EINterval:EVENT<x>:CANBus:DATA:BORDER**

**Function** Sets the byte order of the CAN bus signal trigger data or queries the current setting.

**Syntax** :TRIGger:EINterval:EVENT<x>:CANBus:  
DATA:BORDER {BIG|LITTLE}  
:TRIGger:EINterval:EVENT<x>:CANBus:  
DATA:BORDER?  
<x> = 1 or 2

**Example** :TRIGGER:EINTERVAL:EVENT1:CANBUS:  
DATA:BORDER BIG:  
TRIGGER:EINTERVAL:EVENT1:CANBUS:  
DATA:BORDER? -> :TRIGGER:EINTERVAL:  
EVENT1:CANBUS:DATA:BORDER BIG

**:TRIGger:EINterval:EVENT<x>:CANBus:DATA:CONDition**

**Function** Sets the data condition of the CAN bus signal trigger or queries the current setting.

**Syntax** :TRIGger:EINterval:EVENT<x>:CANBus:  
DATA:CONDition {BETween|DONTcare|  
FALSe|GTHan|LTHan|ORANge|TRUE}  
:TRIGger:EINterval:EVENT<x>:CANBus:  
DATA:CONDition?  
<x> = 1 or 2

**Example** :TRIGGER:EINTERVAL:EVENT1:CANBUS:  
DATA:CONDITION BETWEEN  
:TRIGGER:EINTERVAL:EVENT1:CANBUS:  
DATA:CONDITION? -> TRIGGER:  
EINTERVAL:EVENT1:CANBUS:DATA:  
CONDITION BETWEEN

## 5.29 TRIGger Group

### **:TRIGger:EINterval:EVENT<x>:CANBus:DATA:DATA<x>**

**Function** Sets the comparison data of the CAN bus signal trigger data or queries the current setting.

**Syntax** :TRIGger:EINterval:EVENT<x>:CANBus:DATA:DATA<x> {<Nrf>}  
:TRIGger:EINterval:EVENT<x>:CANBus:DATA:DATA<x>?  
<x> of EVENT<x> = 1 or 2  
<x> of DATA<x> = 1 or 2  
<Nrf> = See the User's Manual (IM701310-01E).

**Example** :TRIGGER:EINTERVAL:EVENT1:CANBUS:DATA:DATA1 1  
:TRIGGER:EINTERVAL:EVENT1:CANBUS:DATA:DATA1? -> :TRIGGER:EINTERVAL:EVENT1:CANBUS:DATA:DATA1 1.0000000E+00

**Description**

- **Use** :TRIGger:EINterval:EVENT<x>:CANBus:DATA:DATA1 **when** :TRIGger:EINterval:EVENT<x>:CANBus:DATA:CONdiTion GTHan **is specified**.
- **Use** :TRIGger:EINterval:EVENT<x>:CANBus:DATA:DATA2 **when** :TRIGger:EINterval:EVENT<x>:CANBus:DATA:CONdiTion LTHan **is specified**.
- **Use** :TRIGger:EINterval:EVENT<x>:CANBus:DATA:DATA1 **to set the smaller value and** :TRIGger:EINterval:EVENT<x>:CANBus:DATA:DATA2 **to set the larger value when** :TRIGger:EINterval:EVENT<x>:CANBus:DATA:CONdiTion BETWeen|ORANge **is specified**.

### **:TRIGger:EINterval:EVENT<x>:CANBus:DATA:DLC**

**Function** Sets the number of valid bytes (DLC) of the CAN bus signal trigger data or queries the current setting.

**Syntax** :TRIGger:EINterval:EVENT<x>:CANBus:DATA:DLC {<Nrf>}  
:TRIGger:EINterval:EVENT<x>:CANBus:DATA:DLC?  
<x> = 1 or 2  
<Nrf> = 0 to 8

**Example** :TRIGGER:EINTERVAL:EVENT1:CANBUS:DATA:DLC 0  
:TRIGGER:EINTERVAL:EVENT1:CANBUS:DATA:DLC? -> :TRIGGER:EINTERVAL:EVENT1:CANBUS:DATA:DLC 0

### **:TRIGger:EINterval:EVENT<x>:CANBus:DATA:HEXA**

**Function** Sets the CAN bus signal trigger data in hexadecimal notation.

**Syntax** :TRIGger:EINterval:EVENT<x>:CANBus:DATA:HEXA {<String>}  
<x> = 1 or 2  
<String> = Up to 16 characters by combining '0' to 'F' and 'X' (in one-byte unit)

**Example** :TRIGGER:EINTERVAL:EVENT1:CANBUS:DATA:HEXA "A9"

### **:TRIGger:EINterval:EVENT<x>:CANBus:DATA:MSBLsb**

**Function** Sets the MSB and LSB bits of the CAN bus signal trigger data or queries the current setting.

**Syntax** :TRIGger:EINterval:EVENT<x>:CANBus:DATA:MSBLsb {<Nrf>,<Nrf>}  
:TRIGger:EINterval:EVENT<x>:CANBus:DATA:MSBLsb?  
<x> = 1 or 2  
<Nrf> = See the User's Manual (IM701310-01E).

**Example** :TRIGGER:EINTERVAL:EVENT1:CANBUS:DATA:MSBLSB 1,0  
:TRIGGER:EINTERVAL:EVENT1:CANBUS:DATA:MSBLSB? -> :TRIGGER:EINTERVAL:EVENT1:CANBUS:DATA:MSBLSB 1,0

### **:TRIGger:EINterval:EVENT<x>:CANBus:DATA:PATtern**

**Function** Sets the CAN bus signal trigger data in binary notation or queries the current setting.

**Syntax** :TRIGger:EINterval:EVENT<x>:CANBus:DATA:PATtern {<String>}  
:TRIGger:EINterval:EVENT<x>:CANBus:DATA:PATtern?  
<x> = 1 or 2  
<String> = Up to 64 characters by combining '0,' '1,' and 'X' (in one-byte unit)

**Example** :TRIGGER:EINTERVAL:EVENT1:CANBUS:DATA:PATTERN "11011111"  
:TRIGGER:EINTERVAL:EVENT1:CANBUS:DATA:PATTERN? -> :TRIGGER:EINTERVAL:EVENT1:CANBUS:DATA:PATTERN "11011111"

**:TRIGger:EINTerval:EVENT<x>:CANBus:DATA:SIGN**

**Function** Sets the sign of the CAN bus signal trigger data or queries the current setting.

**Syntax** :TRIGger:EINTerval:EVENT<x>:CANBus:  
DATA:SIGN {SIGN|UNSign}  
:TRIGger:EINTerval:EVENT<x>:CANBus:  
DATA:SIGN?  
<x> = 1 or 2

**Example** :TRIGGER:EINTERVAL:EVENT1:CANBUS:  
DATA:SIGN SIGN  
:TRIGGER:EINTERVAL:EVENT1:CANBUS:  
DATA:SIGN? -> :TRIGGER:EINTERVAL:  
EVENT1:CANBUS:DATA:SIGN SIGN

**:TRIGger:EINTerval:EVENT<x>:CANBus:IDEXt?**

**Function** Queries all settings related to the ID of the extended format of the CAN bus signal trigger.

**Syntax** :TRIGger:EINTerval:EVENT<x>:CANBus:  
IDEXt?  
<x> = 1 or 2

**Example** :TRIGGER:EINTERVAL:EVENT1:CANBUS:  
IDEXt? -> :TRIGGER:EINTERVAL:EVENT1:  
CANBUS:IDEXt:PATTERN "1100101101110000  
11101110111111"

**:TRIGger:EINTerval:EVENT<x>:CANBus:IDEXt:HEXA**

**Function** Sets the ID of the extended format of the CAN bus signal trigger in hexadecimal notation.

**Syntax** :TRIGger:EINTerval:EVENT<x>:CANBus:  
IDEXt:HEXA {<String>}  
<x> = 1 or 2

<String> = 8 characters by combining '0' to 'F' and 'X'  
**Example** :TRIGGER:EINTERVAL:EVENT1:CANBUS:  
IDEXt:HEXA "1AEF5906"

**:TRIGger:EINTerval:EVENT<x>:CANBus:IDEXt:PATtern**

**Function** Sets the ID of the extended format of the CAN bus signal trigger in binary notation or queries the current setting.

**Syntax** :TRIGger:EINTerval:EVENT<x>:CANBus:  
IDEXt:PATtern {<String>}  
:TRIGger:EINTerval:EVENT<x>:CANBus:  
IDEXt:PATtern?

<x> = 1 or 2  
<String> = 29 characters by combining '0,' '1,' and 'X'  
**Example** :TRIGGER:EINTERVAL:EVENT1:CANBUS:  
IDEXt:PATTERN "1100101101110000111011  
10111111"  
:TRIGGER:EINTERVAL:EVENT1:CANBUS:  
IDEXt:PATTERN? -> :TRIGGER:EINTERVAL:  
EVENT1:CANBUS:IDEXt:PATTERN "11001011  
0111000011101110111111"

## 5.29 TRIGger Group

### :TRIGger:EINterval:EVENT<x>:CANBus:IDOR?

**Function** Queries all settings related to the OR condition of the CAN bus signal trigger.

**Syntax** :TRIGger:EINterval:EVENT<x>:CANBus:IDOR?  
<x> = 1 or 2

**Example** :TRIGGER:EINTERVAL:EVENT1:CANBUS:IDOR? -> :TRIGGER:EINTERVAL:EVENT1:CANBUS:IDOR:ID1:ACK DONTCARE;DATA: BORDER BIG;CONDITION DONTCARE; DATA1 0.0000000E+00; DATA2 255.00000E+00;DLC 8;MSBLSB 7,0; PATTERN "0000000100100011010001010110011100010011010101110011011101110111";SIGN UNSIGN;;TRIGGER:EINTERVAL:EVENT1:CANBUS:IDOR:ID1:FORMAT STD;IDEXT:PATTERN"11010101110011011101110000?";:TRIGGER:EINTERVAL:EVENT1:CANBUS:IDOR:ID1:IDSTD: PATTERN "00100100011";:TRIGGER:EINTERVAL:EVENT1:CANBUS:IDOR:ID1:MODE 0;RTR DATA;;TRIGGER:EINTERVAL:EVENT1:CANBUS:IDOR:ID2:ACK DONTCARE;DATA: BORDER BIG;CONDITION DONTCARE; DATA1 0.0000000E+00; DATA2 255.00000E+00;DLC 8;MSBLSB 7,0; PATTERN ?1111110110111001011101010011000011001000010000?;SIGN UNSIGN;;TRIGGER:EINTERVAL:EVENT1:CANBUS:IDOR:ID2:FORMAT STD;IDEXT:PATTERN"10010001101000101011001111000";:TRIGGER:EINTERVAL:EVENT1:CANBUS:IDOR:ID2:IDSTD: PATTERN "10001010110";:TRIGGER:EINTERVAL:EVENT1:CANBUS:IDOR:ID2:MODE 0;RTR DATA;;TRIGGER:EINTERVAL:EVENT1:CANBUS:IDOR:ID3:ACK DONTCARE;DATA: BORDER BIG;CONDITION DONTCARE; DATA1 0.0000000E+00; DATA2 255.00000E+00;DLC 8.....

### :TRIGger:EINterval:EVENT<x>:CANBus:IDOR:ID<x>?

**Function** Queries all settings related to each ID of the OR condition of the CAN bus signal trigger.

**Syntax** :TRIGger:EINterval:EVENT<x>:CANBus:IDOR:ID<x>?  
<x> of EVENT<x> = 1 or 2  
<x> of ID<x> = 1 to 4

**Example** :TRIGGER:EINTERVAL:EVENT1:CANBUS:IDOR:ID1? -> :TRIGGER:EINTERVAL:EVENT1:CANBUS:IDOR:ID1:ACK DONTCARE;DATA: BORDER BIG;CONDITION DONTCARE; DATA1 0.0000000E+00; DATA2 255.00000E+00;DLC 8;MSBLSB 7,0; PATTERN "0000000100100011010001010110011100010011010101110011011101110111";SIGN UNSIGN;;TRIGGER:EINTERVAL:EVENT1:CANBUS:IDOR:ID1:FORMAT STD;IDEXT:PATTERN "11010101110011011101110011110000";:TRIGGER:EINTERVAL:EVENT1:CANBUS:IDOR:ID1:IDSTD: PATTERN"00100100011";:TRIGGER:EINTERVAL:EVENT1:CANBUS:IDOR:ID1:MODE 0;RTR DATA

### :TRIGger:EINterval:EVENT<x>:CANBus:IDOR:ID<x>:ACK

**Function** Sets each ACK trigger or queries the current setting.

**Syntax** :TRIGger:EINterval:EVENT<x>:CANBus:IDOR:ID<x>:ACK {ACK|ACKBoth|DONTcare|NONack}  
:TRIGger:EINterval:EVENT<x>:CANBus:IDOR:ID<x>:ACK?  
<x> of EVENT<x> = 1 or 2  
<x> of ID<x> = 1 to 4

**Example** :TRIGGER:EINTERVAL:EVENT1:CANBUS:IDOR:ID1:ACK ACK  
:TRIGGER:EINTERVAL:EVENT1:CANBUS:IDOR:ID1:ACK? -> :TRIGGER:EINTERVAL:EVENT1:CANBUS:IDOR:ID1:ACK ACK

**:TRIGger:EINterval:EVENT<x>:CANBus:  
IDOR:ID<x>:DATA?**

**Function** Queries all settings related to each data of the OR condition of the CAN bus signal trigger.

**Syntax** :TRIGger:EINterval:EVENT<x>:CANBus:  
IDOR:ID<x>:DATA?  
<x> of EVENT<x> = 1 or 2  
<x> of ID<x> = 1 to 4

**Example** :TRIGGER:EINTERVAL:EVENT1:CANBUS:  
IDOR:ID1:DATA? -> :TRIGGER:EINTERVAL:  
EVENT1:CANBUS:IDOR:ID1:DATA:  
BORDER BIG;CONDITION DONTCARE;  
DATA1 0.0000000E+00;  
DATA2 255.000000E+00;DLC 8;MSBLSB 7,0;  
PATTERN "0000000100100011010001010110  
011110001001101010111100110111101111"  
;SIGN UNSIGN

**:TRIGger:EINterval:EVENT<x>:CANBus:  
IDOR:ID<x>:DATA:BORDER**

**Function** Sets byte order of each data of the OR condition of the CAN bus signal trigger or queries the current setting.

**Syntax** :TRIGger:EINterval:EVENT<x>:CANBus:  
IDOR:ID<x>:DATA:BORDER {BIG|LITTLE}  
:TRIGger:EINterval:EVENT<x>:CANBus:  
IDOR:ID<x>:DATA:BORDER?  
<x> of EVENT<x> = 1 or 2  
<x> of ID<x> = 1 to 4

**Example** :TRIGGER:EINTERVAL:EVENT1:CANBUS:  
IDOR:ID1:DATA:BORDER BIG  
:TRIGGER:EINTERVAL:EVENT1:CANBUS:  
IDOR:ID1:DATA:BORDER? -> :TRIGGER:  
EINTERVAL:EVENT1:CANBUS:IDOR:ID1:  
DATA:BORDER BIG

**:TRIGger:EINterval:EVENT<x>:CANBus:  
IDOR:ID<x>:DATA:CONDition**

**Function** Sets each data condition of the OR condition of the CAN bus signal trigger or queries the current setting.

**Syntax** :TRIGger:EINterval:EVENT<x>:CANBus:  
IDOR:ID<x>:DATA:CONDition {BETween|  
DONTcare|FALSe|GTHan|LTHan|ORANge|  
TRUE}  
:TRIGger:EINterval:EVENT<x>:CANBus:  
IDOR:ID<x>:DATA:CONDition?  
<x> of EVENT<x> = 1 or 2  
<x> of ID<x> = 1 to 4

**Example** :TRIGGER:EINTERVAL:EVENT1:CANBUS:  
IDOR:ID1:DATA:CONDITION BETWEEN  
:TRIGGER:EINTERVAL:EVENT1:CANBUS:  
IDOR:ID1:DATA:CONDITION? -> :TRIGGER:  
EINTERVAL:EVENT1:CANBUS:IDOR:ID1:  
DATA:CONDITION BETWEEN

**:TRIGger:EINterval:EVENT<x>:CANBus:  
IDOR:ID<x>:DATA:DATA<x>**

**Function** Sets comparison data of each data of the OR condition of the CAN bus signal trigger or queries the current setting.

**Syntax** :TRIGger:EINterval:EVENT<x>:CANBus:  
IDOR:ID<x>:DATA:DATA<x> {<NRf>}  
:TRIGger:EINterval:EVENT<x>:CANBus:  
IDOR:ID<x>:DATA:DATA<x>?  
<x> of EVENT<x> = 1 or 2  
<x> of ID<x> = 1 to 4  
<x> of DATA<x> = 1 or 2  
<NRf> = See the User's Manual (IM701310-01E).

**Example** :TRIGGER:EINTERVAL:EVENT1:CANBUS:  
IDOR:ID1:DATA:DATA1 1  
:TRIGGER:EINTERVAL:EVENT1:CANBUS:  
IDOR:ID1:DATA:DATA1? -> :TRIGGER:  
EINTERVAL:EVENT1:CANBUS:IDOR:ID1:  
DATA:DATA1 1.0000000E+00

**Description** • Use :TRIGger:EINterval:EVENT<x>:  
CANBus:IDOR:ID<x>:DATA:DATA1 when :  
TRIGger:EINterval:EVENT<x>:CANBus:  
IDOR:ID<x>:DATA:CONDition GTHan is  
specified.  
• Use :TRIGger:EINterval:EVENT<x>:  
CANBus:IDOR:ID<x>:DATA:DATA2 when :  
TRIGger:EINterval:EVENT<x>:CANBus:  
IDOR:ID<x>:DATA:CONDition LTHan is  
specified.  
• Use :TRIGger:EINterval:EVENT<x>:  
CANBus:IDOR:ID<x>:DATA:DATA1 to set the  
smaller value and :TRIGger:EINterval:  
EVENT<x>:CANBus:IDOR:ID<x>:DATA:  
DATA2 to set the larger value when :TRIGger:  
EINterval:EVENT<x>:CANBus:IDOR:  
ID<x>:DATA:CONDition BETWEE|ORANge  
is specified.

**:TRIGger:EINterval:EVENT<x>:CANBus:  
IDOR:ID<x>:DATA:DLC**

**Function** Sets the number of valid bytes (DLC) of each data of the OR condition of the CAN bus signal trigger or queries the current setting.

**Syntax** :TRIGger:EINterval:EVENT<x>:CANBus:  
IDOR:ID<x>:DATA:DLC {<NRf>}  
:TRIGger:EINterval:EVENT<x>:CANBus:  
IDOR:ID<x>:DATA:DLC?  
<x> of EVENT<x> = 1 or 2  
<x> of ID<x> = 1 to 4  
<NRf> = 0 to 8

**Example** :TRIGGER:EINTERVAL:EVENT1:CANBUS:  
IDOR:ID1:DATA:DLC 0  
:TRIGGER:EINTERVAL:EVENT1:CANBUS:  
IDOR:ID1:DATA:DLC? -> :TRIGGER:  
EINTERVAL:EVENT1:CANBUS:IDOR:ID1:  
DATA:DLC 0

## 5.29 TRIGger Group

**:TRIGger:EINterval:EVENT<x>:CANBus:  
IDOR:ID<x>:DATA:HEXA**

**Function** Sets each data of the OR condition of the CAN bus signal trigger in hexadecimal notation.

**Syntax** :TRIGger:EINterval:EVENT<x>:CANBus:  
IDOR:ID<x>:DATA:HEXA {<String>}  
<x> of EVENT<x> = 1 or 2  
<x> of ID<x> = 1 to 4  
<String> = Up to 16 characters by combining '0' to 'F' and 'X' (in one-byte unit)

**Example** :TRIGGER:EINTERVAL:EVENT1:CANBUS:  
IDOR:ID1:DATA:HEXA "A9"

**:TRIGger:EINterval:EVENT<x>:CANBus:  
IDOR:ID<x>:DATA:MSBLSb**

**Function** Sets the MSB and LSB bits of each data of the OR condition of the CAN bus signal trigger or queries the current setting.

**Syntax** :TRIGger:EINterval:EVENT<x>:CANBus:  
IDOR:ID<x>:DATA:MSBLSb {<NRf>,<NRf>}  
:TRIGger:EINterval:EVENT<x>:CANBus:  
IDOR:ID<x>:DATA:MSBLSb?  
<x> of EVENT<x> = 1 or 2  
<x> of ID<x> = 1 to 4  
<NRf> = See the User's Manual (IM701310-01E).

**Example** :TRIGGER:EINTERVAL:EVENT1:CANBUS:  
IDOR:ID1:DATA:MSBLSB 1,0  
:TRIGGER:EINTERVAL:EVENT1:CANBUS:  
IDOR:ID1:DATA:MSBLSB? -> :TRIGGER:  
EINTERVAL:EVENT1:CANBUS:IDOR:ID1:  
DATA:MSBLSB 1,0

**:TRIGger:EINterval:EVENT<x>:CANBus:  
IDOR:ID<x>:DATA:PATtern**

**Function** Sets each data of the OR condition of the CAN bus signal trigger in binary notation or queries the current setting.

**Syntax** :TRIGger:EINterval:EVENT<x>:CANBus:  
IDOR:ID<x>:DATA:PATtern {<String>}  
:TRIGger:EINterval:EVENT<x>:CANBus:  
IDOR:ID<x>:DATA:PATtern?  
<x> of EVENT<x> = 1 or 2  
<x> of ID<x> = 1 to 4  
<String> = Up to 64 characters by combining '0,' '1,' and 'X' (in one-byte unit)

**Example** :TRIGGER:EINTERVAL:EVENT1:CANBUS:  
IDOR:ID1:DATA:PATTERN "11011111"  
:TRIGGER:EINTERVAL:EVENT1:CANBUS:  
IDOR:ID1:DATA:PATTERN? -> :TRIGGER:  
EINTERVAL:EVENT1:CANBUS:IDOR:ID1:  
DATA:PATTERN "11011111"

**:TRIGger:EINterval:EVENT<x>:CANBus:  
IDOR:ID<x>:DATA:SIGN**

**Function** Sets sign of each data of the OR condition of the CAN bus signal trigger or queries the current setting.

**Syntax** :TRIGger:EINterval:EVENT<x>:CANBus:  
IDOR:ID<x>:DATA:SIGN {SIGN|UNSign}  
:TRIGger:EINterval:EVENT<x>:CANBus:  
IDOR:ID<x>:DATA:SIGN?  
<x> of EVENT<x> = 1 or 2  
<x> of ID<x> = 1 to 4

**Example** :TRIGGER:EINTERVAL:EVENT1:CANBUS:  
IDOR:ID1:DATA:SIGN SIGN  
:TRIGGER:EINTERVAL:EVENT1:CANBUS:  
IDOR:ID1:DATA:SIGN? -> :TRIGGER:  
EINTERVAL:EVENT1:CANBUS:IDOR:ID1:  
DATA:SIGN SIGN

**:TRIGger:EINterval:EVENT<x>:CANBus:  
IDOR:ID<x>:FORMat**

**Function** Sets each message format (standard or extended) of the OR condition of the CAN bus signal trigger or queries the current setting.

**Syntax** :TRIGger:EINterval:EVENT<x>:CANBus:  
IDOR:ID<x>:FORMat {STD|EXT}  
:TRIGger:EINterval:EVENT<x>:CANBus:  
IDOR:ID<x>:FORMat?  
<x> of EVENT<x> = 1 or 2  
<x> of ID<x> = 1 to 4

**Example** :TRIGGER:EINTERVAL:EVENT1:CANBUS:  
IDOR:ID1:FORMAT STD  
:TRIGGER:EINTERVAL:EVENT1:CANBUS:  
IDOR:ID1:FORMAT? -> :TRIGGER:  
EINTERVAL:EVENT1:CANBUS:IDOR:ID1:  
FORMAT STD

**:TRIGger:EINterval:EVENT<x>:CANBus:  
IDOR:ID<x>:IDEXt?**

**Function** Queries all settings related to the ID of each extended format of the OR condition of the CAN bus signal trigger.

**Syntax** :TRIGger:EINterval:EVENT<x>:CANBus:  
IDOR:ID<x>:IDEXt?  
<x> of EVENT<x> = 1 or 2  
<x> of ID<x> = 1 to 4

**Example** :TRIGGER:EINTERVAL:EVENT1:CANBUS:  
IDOR:ID1:IDEXT? -> :TRIGGER:  
EINTERVAL:EVENT1:CANBUS:IDOR:ID1:  
IDEXT:PATTERN "1100101101110000111011  
1011111"



**:TRIGger:EINterval:EVENT<x>:CANBus:  
IDOR:ID<x>:IDEXt:HEXA**

Function Sets the ID of each extended format of the OR condition of the CAN bus signal trigger in hexadecimal notation.

Syntax :TRIGger:EINterval:EVENT<x>:CANBus:  
IDOR:ID<x>:IDEXt:HEXA {<String>}  
<x> of EVENT<x> = 1 or 2  
<x> of ID<x> = 1 to 4  
<String> = 8 characters by combining '0' to 'F' and 'X'

Example :TRIGGER:EINTERVAL:EVENT1:CANBUS:  
IDOR:ID1:IDEXT:HEXA "1AEF5906"

**:TRIGger:EINterval:EVENT<x>:CANBus:  
IDOR:ID<x>:IDEXt:PATtern**

Function Sets the ID of each extended format of the OR condition of the CAN bus signal trigger in binary notation or queries the current setting.

Syntax :TRIGger:EINterval:EVENT<x>:CANBus:  
IDOR:ID<x>:IDEXt:PATtern {<String>}  
:TRIGger:EINterval:EVENT<x>:CANBus:  
IDOR:ID<x>:IDEXt:PATtern?  
<x> of EVENT<x> = 1 or 2  
<x> of ID<x> = 1 to 4  
<String> = 29 characters by combining '0','1,' and 'X'

Example :TRIGGER:EINTERVAL:EVENT1:CANBUS:  
IDOR:ID1:IDEXT:PATTERN "1100101101110  
0001110111011111"  
:TRIGGER:EINTERVAL:EVENT1:CANBUS:  
IDOR:ID1:IDEXT:PATTERN? -> :TRIGGER:  
EINTERVAL:EVENT1:CANBUS:IDOR:ID1:  
IDEXT:PATTERN "1100101101110000111011  
1011111"

**:TRIGger:EINterval:EVENT<x>:CANBus:  
IDOR:ID<x>:IDSTd?**

Function Queries all settings related to the ID of each standard format of the OR condition of the CAN bus signal trigger.

Syntax :TRIGger:EINterval:EVENT<x>:CANBus:  
IDOR:ID<x>:IDSTd?  
<x> of EVENT<x> = 1 or 2  
<x> of ID<x> = 1 to 4

Example :TRIGGER:EINTERVAL:EVENT1:CANBUS:  
IDOR:ID1:IDSTD? -> :TRIGGER:  
EINTERVAL:EVENT1:CANBUS:IDOR:ID1:  
IDSTD:PATTERN "00011111101"

**:TRIGger:EINterval:EVENT<x>:CANBus:  
IDOR:ID<x>:IDSTd:HEXA**

Function Sets the ID of each standard format of the OR condition of the CAN bus signal trigger in hexadecimal notation.

Syntax :TRIGger:EINterval:EVENT<x>:CANBus:  
IDOR:ID<x>:IDSTd:HEXA {<String>}  
<x> of EVENT<x> = 1 or 2  
<x> of ID<x> = 1 to 4  
<String> = 3 characters by combining '0' to 'F' and 'X'

Example :TRIGGER:EINTERVAL:EVENT1:CANBUS:  
IDOR:ID1:IDSTD:HEXA "5DF"

**:TRIGger:EINterval:EVENT<x>:CANBus:  
IDOR:ID<x>:IDSTd:PATtern**

Function Sets the ID of each standard format of the OR condition of the CAN bus signal trigger in binary notation or queries the current setting.

Syntax :TRIGger:EINterval:EVENT<x>:CANBus:  
IDOR:ID<x>:IDSTd:PATtern {<String>}  
:TRIGger:EINterval:EVENT<x>:CANBus:  
IDOR:ID<x>:IDSTd:PATtern?  
<x> of EVENT<x> = 1 or 2  
<x> of ID<x> = 1 to 4  
<String> = 11 characters by combining '0','1,' and 'X'

Example :TRIGGER:EINTERVAL:EVENT1:CANBUS:  
IDOR:ID1:IDSTD:PATTERN "10111011111"  
:TRIGGER:EINTERVAL:EVENT1:CANBUS:  
IDOR:ID1:IDSTD:PATTERN? -> :TRIGGER:  
EINTERVAL:EVENT1:CANBUS:IDOR:ID1:  
IDSTD:PATTERN "10111011111"

**:TRIGger:EINterval:EVENT<x>:CANBus:  
IDOR:ID<x>:MODE**

Function Enables or disables each condition of the OR condition of the CAN bus signal trigger or queries the current setting.

Syntax :TRIGger:EINterval:EVENT<x>:CANBus:  
IDOR:ID<x>:MODE {<Boolean>}  
:TRIGger:EINterval:EVENT<x>:CANBus:  
IDOR:ID<x>:MODE?  
<x> of EVENT<x> = 1 or 2  
<x> of ID<x> = 1 to 4

Example :TRIGGER:EINTERVAL:EVENT1:CANBUS:  
IDOR:ID1:MODE ON  
:TRIGGER:EINTERVAL:EVENT1:CANBUS:  
IDOR:ID1:MODE? -> :TRIGGER:EINTERVAL:  
EVENT1:CANBUS:IDOR:ID1:MODE 1

## 5.29 TRIGger Group

### **:TRIGger:EINTerval:EVENT<x>:CANBus:IDOR:ID<x>:RTR**

**Function** Sets each RTR of the OR condition of the CAN bus signal trigger or queries the current setting.

**Syntax** :TRIGger:EINTerval:EVENT<x>:CANBus:IDOR:ID<x>:RTR {DATA|DONTcare|REMOte}  
:TRIGger:EINTerval:EVENT<x>:CANBus:IDOR:ID<x>:RTR?  
<x> of EVENT<x> = 1 or 2  
<x> of ID<x> = 1 to 4

**Example** :TRIGGER:EINTERVAL:EVENT1:CANBUS:IDOR:ID1:RTR DATA  
:TRIGGER:EINTERVAL:EVENT1:CANBUS:IDOR:ID1:RTR? -> :TRIGGER:EINTERVAL:EVENT1:CANBUS:IDOR:ID1:RTR DATA

### **:TRIGger:EINTerval:EVENT<x>:CANBus:IDSTd?**

**Function** Queries all settings related to the ID of the standard format of the CAN bus signal trigger.

**Syntax** :TRIGger:EINTerval:EVENT<x>:CANBus:IDSTd?  
<x> = 1 or 2

**Example** :TRIGGER:EINTERVAL:EVENT1:CANBUS:IDSTD? -> :TRIGGER:EINTERVAL:EVENT1:CANBUS:IDSTD:PATTERN "00011111101"

### **:TRIGger:EINTerval:EVENT<x>:CANBus:IDSTd:HEXA**

**Function** Sets the ID of the standard format of the CAN bus signal trigger in hexadecimal notation.

**Syntax** :TRIGger:EINTerval:EVENT<x>:CANBus:IDSTd:HEXA {<String>}  
<x> = 1 or 2  
<String> = 8 characters by combining '0' to 'F' and 'X'

**Example** :TRIGGER:EINTERVAL:EVENT1:CANBUS:IDSTD:HEXA "5DF"

### **:TRIGger:EINTerval:EVENT<x>:CANBus:IDSTd:PATtern**

**Function** Sets the ID of the standard format of the CAN bus signal trigger in binary notation or queries the current setting.

**Syntax** :TRIGger:EINTerval:EVENT<x>:CANBus:IDSTd:PATtern {<String>}  
:TRIGger:EINTerval:EVENT<x>:CANBus:IDSTd:PATtern?  
<x> = 1 or 2  
<String> = 11 characters by combining '0', '1,' and 'X'

**Example** :TRIGGER:EINTERVAL:EVENT1:CANBUS:IDSTD:PATTERN "10111011111"  
:TRIGGER:EINTERVAL:EVENT1:CANBUS:IDSTD:PATTERN? -> :TRIGGER:EINTERVAL:EVENT1:CANBUS:IDSTD:PATTERN "10111011111"

### **:TRIGger:EINTerval:EVENT<x>:CANBus:MODE**

**Function** Sets the CAN bus signal trigger mode or queries the current setting.

**Syntax** :TRIGger:EINTerval:EVENT<x>:CANBus:MODE {EFrame|IDExt|IDOR|IDSTd|SOF}  
:TRIGger:EINTerval:EVENT<x>:CANBus:MODE?  
<x> = 1 or 2

**Example** :TRIGGER:EINTERVAL:EVENT1:CANBUS:MODE EFRAME  
:TRIGGER:EINTERVAL:EVENT1:CANBUS:MODE? -> :TRIGGER:EINTERVAL:EVENT1:CANBUS:MODE EFRAME

### **:TRIGger:EINTerval:EVENT<x>:CANBus:REcessive**

**Function** Sets the recessive level (bus level) of the CAN bus signal trigger or queries the current setting.

**Syntax** :TRIGger:EINTerval:EVENT<x>:CANBus:REcessive {HIGH|LOW}  
:TRIGger:EINTerval:EVENT<x>:CANBus:REcessive?  
<x> = 1 or 2

**Example** :TRIGGER:EINTERVAL:EVENT1:CANBUS:RECESSIVE HIGH  
:TRIGGER:EINTERVAL:EVENT1:CANBUS:RECESSIVE? -> :TRIGGER:EINTERVAL:EVENT1:CANBUS:RECESSIVE HIGH

**:TRIGger:EINterval:EVENT<x>:CANBus:RTR**  
 Function Sets the RTR of the CAN bus signal trigger or queries the current setting.

Syntax :TRIGger:EINterval:EVENT<x>:CANBus:  
 RTR {DATA|DONTcare|REMOte}  
 :TRIGger:EINterval:EVENT<x>:CANBus:  
 RTR?  
 <x> = 1 or 2

Example :TRIGGER:EINTERVAL:EVENT1:CANBUS:  
 RTR DATA  
 :TRIGGER:EINTERVAL:EVENT1:CANBUS:  
 RTR? -> :TRIGGER:EINTERVAL:EVENT1:  
 CANBUS:RTR DATA

**:TRIGger:EINterval:EVENT<x>:CANBus:SOURCE**

Function Sets the trigger source of the CAN bus signal trigger or queries the current setting.

Syntax :TRIGger:EINterval:EVENT<x>:CANBus:  
 SOURCE {<NRf>}  
 :TRIGger:EINterval:EVENT<x>:CANBus:  
 SOURCE?  
 <x> = 1 or 2  
 <NRf> = 1 to 4

Example :TRIGGER:EINTERVAL:EVENT1:CANBUS:  
 SOURCE 1  
 :TRIGGER:EINTERVAL:EVENT1:CANBUS:  
 SOURCE? -> :TRIGGER:EINTERVAL:EVENT1:  
 CANBUS:SOURCE 1

**:TRIGger:EINterval:EVENT<x>:CANBus:SPOint**

Function Sets the sample point of the CAN bus signal trigger or queries the current setting.

Syntax :TRIGger:EINterval:EVENT<x>:CANBus:  
 SPOint {<NRf>}  
 :TRIGger:EINterval:EVENT<x>:CANBus:  
 SPOint?  
 <x> = 1 or 2  
 <NRf> = 18.8 to 90.6(%)

Example :TRIGGER:EINTERVAL:EVENT1:CANBUS:  
 SPOINT 18.8  
 :TRIGGER:EINTERVAL:EVENT1:CANBUS:  
 SPOINT? -> :TRIGGER:EINTERVAL:EVENT1:  
 CANBUS:SPOINT 18.8E+00

**:TRIGger:EINterval:EVENT<x>:CLOCK?**

Function Queries all settings related to the clock channel of the event.

Syntax :TRIGger:EINterval:EVENT<x>:CLOCK?  
 <x> = 1 or 2

Example :TRIGGER:EINTERVAL:EVENT1:CLOCK?  
 -> :TRIGGER:EINTERVAL:EVENT1:CLOCK:  
 SOURCE 1;POLARITY FALL

**:TRIGger:EINterval:EVENT<x>:CLOCK:POLarity**

Function Sets the polarity of the clock channel of the event or queries the current setting.

Syntax :TRIGger:EINterval:EVENT<x>:CLOCK:  
 POLarity {ENTER|EXIT|FALL|RISE}  
 :TRIGger:EINterval:EVENT<x>:CLOCK:  
 POLarity?  
 <x> = 1 or 2

Example :TRIGGER:EINTERVAL:EVENT1:CLOCK:  
 POLARITY FALL  
 :TRIGGER:EINTERVAL:EVENT1:CLOCK:  
 POLARITY? -> :TRIGGER:EINTERVAL:EVENT1:  
 CLOCK:POLARITY FALL

Description • This command is invalid if :TRIGger:  
 EINterval:EVENT<x>:CLOCK:SOURCE NONE,  
 • {ENTER|EXIT} is valid when :TRIGger:SOURCE:  
 CHANnel<x>:WINDOW ON. For all other cases,  
 {FALL|RISE} is valid.  
 • This command is valid when :TRIGger:  
 EINterval:EVENT<x>:TYPE PSTATE|STATE.

**:TRIGger:EINterval:EVENT<x>:CLOCK:SOURCE**

Function Sets the source waveform of the clock channel of the event or queries the current setting.

Syntax :TRIGger:EINterval:EVENT<x>:CLOCK:  
 SOURCE {<NRf>|NONE}  
 :TRIGger:EINterval:EVENT<x>:CLOCK:  
 SOURCE?  
 <x> = 1 or 2  
 <NRf> = 1 to 4

Example :TRIGGER:EINTERVAL:EVENT1:CLOCK:  
 SOURCE 1  
 :TRIGGER:EINTERVAL:EVENT1:CLOCK:SOURCE?  
 -> :TRIGGER:EINTERVAL:EVENT1:CLOCK:  
 SOURCE 1

Description This command is valid when :TRIGger:EINterval:  
 EVENT<x>:TYPE PSTATE|STATE.

**:TRIGger:EINterval:EVENT<x>:ESTate?**

Function Queries all settings related to the edge/state trigger.

Syntax :TRIGger:EINterval:EVENT<x>:ESTate?  
 <x> = 1 or 2

Example :TRIGGER:EINTERVAL:EVENT1:ESTATE?  
 -> :TRIGGER:EINTERVAL:EVENT1:ESTATE:  
 SOURCE 1;POLARITY FALL

## 5.29 TRIGger Group

### **:TRIGger:EIInterval:EVENT<x>:ESTate:POLarity**

**Function** Sets the polarity of the edge/state trigger or queries the current setting.

**Syntax** :TRIGger:EIInterval:EVENT<x>:ESTate:POLarity {ENTER|EXIT|FALL|RISE}  
:TRIGger:EIInterval:EVENT<x>:ESTate:POLarity?  
<x> = 1 or 2

**Example** :TRIGGER:EIINTERVAL:EVENT1:ESTATE:POLARITY ENTER  
:TRIGGER:EIINTERVAL:EVENT1:ESTATE:POLARITY? -> :TRIGGER:EIINTERVAL:EVENT1:ESTATE:POLARITY ENTER

**Description** • This command is valid when :TRIGger:EIInterval:EVENT<x>:TYPE EDGE and :TRIGger:EIInterval:EVENT<x>:ESTate:SOURce LINE.

- {ENTER|EXIT} is valid when :TRIGger:EIInterval:EVENT<x>:TYPE EDGE|EQUalify and :TRIGger:SOURce:CHANnel<x>:WINDow ON. {FALL|RISE} is valid when :TRIGger:EIInterval:EVENT<x>:TYPE EDGE|EQUalify and :TRIGger:SOURce:CHANnel<x>:WINDow OFF.
- {ENTER|EXIT} is valid when :TRIGger:EIInterval:EVENT<x>:TYPE STATE.

### **:TRIGger:EIInterval:EVENT<x>:ESTate:SOURce**

**Function** Sets the trigger source of the edge/state trigger or queries the current setting.

**Syntax** :TRIGger:EIInterval:EVENT<x>:ESTate:SOURce {<Nrf>|EXternal|LINE}  
:TRIGger:EIInterval:EVENT<x>:ESTate:SOURce?  
<x> = 1 or 2  
<Nrf> = 1 to 4

**Example** :TRIGGER:EIINTERVAL:EVENT1:ESTATE:SOURCE 1  
:TRIGGER:EIINTERVAL:EVENT1:ESTATE:SOURCE? -> :TRIGGER:EIINTERVAL:EVENT1:ESTATE:SOURCE 1

**Description** • This command is valid when :TRIGger:EIInterval:EVENT<x>:TYPE EDGE|EQUalify.

- {<Nrf>|EXternal|LINE} is valid when :TRIGger:EIInterval:EVENT<x>:TYPE EDGE.
- {<Nrf>|EXternal} is valid when :TRIGger:EIInterval:EVENT<x>:TYPE EQUalify.

### **:TRIGger:EIInterval:EVENT<x>:I2Cbus?**

**Function** Queries all settings related to the I<sup>2</sup>C bus trigger of the event.

**Syntax** :TRIGger:EIInterval:EVENT<x>:I2Cbus?<x> = 1 or 2

**Example** :TRIGGER:EIINTERVAL:EVENT1:I2CBUS? -> :TRIGGER:EIINTERVAL:EVENT1:I2CBUS:ADATA:BIT10ADDRESS:PATTERN "10111011111"; :TRIGGER:EIINTERVAL:EVENT1:I2CBUS:ADATA:BIT7ADDRESS:PATTERN "11011110"; :TRIGGER:EIINTERVAL:EVENT1:I2CBUS:ADATA:BIT7APSUB:ADDRESS:PATTERN "10101011"; :TRIGGER:EIINTERVAL:EVENT1:I2CBUS:ADATA:BIT7APSUB:SADDRESS:PATTERN "10101011"; :TRIGGER:EIINTERVAL:EVENT1:I2CBUS:ADATA:TYPE BIT10ADDRESS; :TRIGGER:EIINTERVAL:EVENT1:I2CBUS:CLOCK:SOURCE 1; :TRIGGER:EIINTERVAL:EVENT1:I2CBUS:DATA:BYTE 1; CONDITION TRUE;DPOSITION 1;MODE 1; PATTERN1 "10101011"; PATTERN2 "10101010"; PATTERN3 "10101111"; PATTERN4 "10101011";PMODE DONTCARE; SOURCE 1; :TRIGGER:EIINTERVAL:EVENT1:I2CBUS:GCALL:BIT7MADDRESS:PATTERN "1010101"; :TRIGGER:EIINTERVAL:EVENT1:I2CBUS:GCALL:SBYTE BIT7MADDRESS; :TRIGGER:EIINTERVAL:EVENT1:I2CBUS:MODE ADATA;NAIGNORE: HSMODE 1;RACCESS 1;SBYTE 1; :TRIGGER:EIINTERVAL:EVENT1:I2CBUS:SBHSMODE:TYPE HSMODE

### **:TRIGger:EIInterval:EVENT<x>:I2Cbus:ADATa?**

**Function** Queries all settings related to the address of the I<sup>2</sup>C bus trigger.

**Syntax** :TRIGger:EIInterval:EVENT<x>:I2Cbus:ADATa?<x> = 1 or 2

**Example** :TRIGGER:EIINTERVAL:EVENT1:I2CBUS:ADATa? -> :TRIGGER:EIINTERVAL:EVENT1:I2CBUS:ADATA:BIT10ADDRESS:PATTERN "10111011111"; :TRIGGER:EIINTERVAL:EVENT1:I2CBUS:ADATA:BIT7ADDRESS:PATTERN "11011110"; :TRIGGER:EIINTERVAL:EVENT1:I2CBUS:ADATA:BIT7APSUB:ADDRESS:PATTERN "10101011"; :TRIGGER:EIINTERVAL:EVENT1:I2CBUS:ADATA:BIT7APSUB:SADDRESS:PATTERN "10101011"; :TRIGGER:EIINTERVAL:EVENT1:I2CBUS:ADATA:TYPE BIT10ADDRESS

**:TRIGger:EINterval:EVENT<x>:I2Cbus:ADATa:BIT10address?**

Function Queries all settings related to the 10-bit address of the I<sup>2</sup>C bus trigger.

Syntax :TRIGger:EINterval:EVENT<x>:I2Cbus:ADATa:BIT10address?  
<x> = 1 or 2

Example :TRIGGER:EINTERVAL:EVENT1:I2CBUS:ADATA:BIT10ADDRESS? -> :TRIGGER:EINTERVAL:EVENT1:I2CBUS:ADATA:BIT10ADDRESS:PATTERN "10111011111"

**:TRIGger:EINterval:EVENT<x>:I2Cbus:ADATa:BIT10address:HEXA**

Function Sets the 10-bit address of the I<sup>2</sup>C bus trigger in hexadecimal notation.

Syntax :TRIGger:EINterval:EVENT<x>:I2Cbus:ADATa:BIT10address:HEXA {<String>}  
<x> = 1 or 2  
<String> = 3 characters by combining '0' to 'F' and 'X' (bit 8 is the R/W bit)

Example :TRIGGER:EINTERVAL:EVENT1:I2CBUS:ADATA:BIT10ADDRESS:HEXA "7AB"

**:TRIGger:EINterval:EVENT<x>:I2Cbus:ADATa:BIT10address:PATtern**

Function Sets the 10-bit address of the I<sup>2</sup>C bus trigger in binary notation or queries the current setting.

Syntax :TRIGger:EINterval:EVENT<x>:I2Cbus:ADATa:BIT10address:PATtern {<String>}  
:TRIGger:EINterval:EVENT<x>:I2Cbus:ADATa:BIT10address:PATtern?  
<x> = 1 or 2  
<String> = 11 characters by combining '0', '1', and 'X' (bit 8 is the R/W\_bit)

Example :TRIGGER:EINTERVAL:EVENT1:I2CBUS:ADATA:BIT10ADDRESS:PATTERN "10111011111"  
:TRIGGER:EINTERVAL:EVENT1:I2CBUS:ADATA:BIT10ADDRESS:PATTERN? -> :TRIGGER:EINTERVAL:EVENT1:I2CBUS:ADATA:BIT10ADDRESS:PATTERN "10111011111"

**:TRIGger:EINterval:EVENT<x>:I2Cbus:ADATa:BIT7Address?**

Function Queries all settings related to the 7-bit address of the I<sup>2</sup>C bus trigger.

Syntax :TRIGger:EINterval:EVENT<x>:I2Cbus:ADATa:BIT7Address?  
<x> = 1 or 2

Example :TRIGGER:EINTERVAL:EVENT1:I2CBUS:ADATA:BIT7ADDRESS? -> :TRIGGER:EINTERVAL:EVENT1:I2CBUS:ADATA:BIT7ADDRESS:PATTERN "11011110"

**:TRIGger:EINterval:EVENT<x>:I2Cbus:ADATa:BIT7Address:HEXA**

Function Sets the 7-bit address of the I<sup>2</sup>C bus trigger in hexadecimal notation.

Syntax :TRIGger:EINterval:EVENT<x>:I2Cbus:ADATa:BIT7Address:HEXA {<String>}  
<x> = 1 or 2

<String> = 2 characters by combining '0' to 'F' and 'X' (bit 0 is the R/W bit)

Example :TRIGGER:EINTERVAL:EVENT1:I2CBUS:ADATA:BIT7ADDRESS:HEXA "DE"

**:TRIGger:EINterval:EVENT<x>:I2Cbus:ADATa:BIT7Address:PATtern**

Function Sets the 7-bit address of the I<sup>2</sup>C bus trigger in binary notation or queries the current setting.

Syntax :TRIGger:EINterval:EVENT<x>:I2Cbus:ADATa:BIT7Address:PATtern {<String>}  
:TRIGger:EINterval:EVENT<x>:I2Cbus:ADATa:BIT7Address:PATtern?  
<x> = 1 or 2

<String> = 8 characters by combining '0', '1', and 'X' (bit 0 is the R/W bit)

Example :TRIGGER:EINTERVAL:EVENT1:I2CBUS:ADATA:BIT7ADDRESS:PATTERN "11011110"  
:TRIGGER:EINTERVAL:EVENT1:I2CBUS:ADATA:BIT7ADDRESS:PATTERN? -> :TRIGGER:EINTERVAL:EVENT1:I2CBUS:ADATA:BIT7ADDRESS:PATTERN "11011110"

**:TRIGger:EINterval:EVENT<x>:I2Cbus:ADATa:BIT7APsub?**

Function Queries all settings related to the 7-bit + Sub address of the I<sup>2</sup>C bus trigger.

Syntax :TRIGger:EINterval:EVENT<x>:I2Cbus:ADATa:BIT7APsub?  
<x> = 1 or 2

Example :TRIGGER:EINTERVAL:EVENT1:I2CBUS:ADATA:BIT7APSUB? -> :TRIGGER:EINTERVAL:EVENT1:I2CBUS:ADATA:BIT7APSUB:ADDRESS:PATTERN "10101011"; :TRIGGER:EINTERVAL:EVENT1:I2CBUS:ADATA:BIT7APSUB:SADDRESS:PATTERN "10101011"

## 5.29 TRIGger Group

### **:TRIGger:EINterval:EVENT<x>:I2Cbus:ADATa:BIT7APsub:ADDRESS?**

**Function** Queries all settings related to the 7-bit address of the 7-bit + Sub address of the I<sup>2</sup>C bus trigger.

**Syntax** :TRIGger:EINterval:EVENT<x>:I2Cbus:ADATa:BIT7APsub:ADDRESS?  
<x> = 1 or 2

**Example** :TRIGGER:EINTERVAL:EVENT1:I2CBUS:ADATA:BIT7APSUB:ADDRESS? -> :TRIGGER:EINTERVAL:EVENT1:I2CBUS:ADATA:BIT7APSUB:ADDRESS:PATTERN "10101011"

### **:TRIGger:EINterval:EVENT<x>:I2Cbus:ADATa:BIT7APsub:ADDRESS:HEXA**

**Function** Sets the 7-bit address of the 7-bit + Sub address of the I<sup>2</sup>C bus trigger in hexadecimal notation.

**Syntax** :TRIGger:EINterval:EVENT<x>:I2Cbus:ADATa:BIT7APsub:ADDRESS:HEXA {<String>}  
<x> = 1 or 2  
<String> = 2 characters by combining '0' to 'F' and 'X' (bit 0 is the R/W bit)

**Example** :TRIGGER:EINTERVAL:EVENT1:I2CBUS:ADATA:BIT7APSUB:ADDRESS:HEXA "AB"

### **:TRIGger:EINterval:EVENT<x>:I2Cbus:ADATa:BIT7APsub:ADDRESS:PATtern**

**Function** Sets the 7-bit address of the 7-bit + Sub address of the I<sup>2</sup>C bus trigger in binary notation or queries the current setting.

**Syntax** :TRIGger:EINterval:EVENT<x>:I2Cbus:ADATa:BIT7APsub:ADDRESS:PATtern {<String>}  
:TRIGger:EINterval:EVENT<x>:I2Cbus:ADATa:BIT7APsub:ADDRESS:PATtern?  
<x> = 1 or 2  
<String> = 8 characters by combining '0', '1', and 'X' (bit 0 is the R/W bit)

**Example** :TRIGGER:EINTERVAL:EVENT1:I2CBUS:ADATA:BIT7APSUB:ADDRESS:PATTERN "10101011"  
:TRIGGER:EINTERVAL:EVENT1:I2CBUS:ADATA:BIT7APSUB:ADDRESS:PATTERN? -> :TRIGGER:EINTERVAL:EVENT1:I2CBUS:ADATA:BIT7APSUB:ADDRESS:PATTERN "10101011"

### **:TRIGger:EINterval:EVENT<x>:I2Cbus:ADATa:BIT7APsub:SADDRESS?**

**Function** Queries all settings related to the Sub address of the 7-bit + Sub address of the I<sup>2</sup>C bus trigger.

**Syntax** :TRIGger:EINterval:EVENT<x>:I2Cbus:ADATa:BIT7APsub:SADDRESS?  
<x> = 1 or 2

**Example** :TRIGGER:EINTERVAL:EVENT1:I2CBUS:ADATA:BIT7APSUB:SADDRESS? -> :TRIGGER:EINTERVAL:EVENT1:I2CBUS:ADATA:BIT7APSUB:SADDRESS:PATTERN "10101011"

### **:TRIGger:EINterval:EVENT<x>:I2Cbus:ADATa:BIT7APsub:SADDRESS:HEXA**

**Function** Sets the Sub address of the 7-bit + Sub address of the I<sup>2</sup>C bus trigger in hexadecimal notation.

**Syntax** :TRIGger:EINterval:EVENT<x>:I2Cbus:ADATa:BIT7APsub:SADDRESS:HEXA {<String>}  
<x> = 1 or 2  
<String> = 2 characters by combining '0' to 'F' and 'X'

**Example** :TRIGGER:EINTERVAL:EVENT1:I2CBUS:ADATA:BIT7APSUB:SADDRESS:HEXA "EF"

### **:TRIGger:EINterval:EVENT<x>:I2Cbus:ADATa:BIT7APsub:SADDRESS:PATtern**

**Function** Sets the Sub address of the 7-bit + Sub address of the I<sup>2</sup>C bus trigger in binary notation or queries the current setting.

**Syntax** :TRIGger:EINterval:EVENT<x>:I2Cbus:ADATa:BIT7APsub:SADDRESS:PATtern {<String>}  
:TRIGger:EINterval:EVENT<x>:I2Cbus:ADATa:BIT7APsub:SADDRESS:PATtern?  
<x> = 1 or 2  
<String> = 8 characters by combining '0', '1', and 'X'

**Example** :TRIGGER:EINTERVAL:EVENT1:I2CBUS:ADATA:BIT7APSUB:SADDRESS:PATTERN "10101011"  
:TRIGGER:EINTERVAL:EVENT1:I2CBUS:ADATA:BIT7APSUB:SADDRESS:PATTERN?  
-> :TRIGGER:EINTERVAL:EVENT1:I2CBUS:ADATA:BIT7APSUB:SADDRESS:PATTERN "10101011"

### **:TRIGger:EINterval:EVENT<x>:I2Cbus:ADATa:TYPE**

**Function** Sets the address type of the I<sup>2</sup>C bus trigger or queries the current setting.

**Syntax** :TRIGger:EINterval:EVENT<x>:I2Cbus:ADATa:TYPE {BIT10address|BIT7Address|BIT7APsub}  
:TRIGger:EINterval:EVENT<x>:I2Cbus:ADATa:TYPE?  
<x> = 1 or 2

**Example** :TRIGGER:EINTERVAL:EVENT1:I2CBUS:ADATA:TYPE BIT10ADDRESS  
:TRIGGER:EINTERVAL:EVENT1:I2CBUS:ADATA:TYPE? -> :TRIGGER:EINTERVAL:EVENT1:I2CBUS:ADATA:TYPE BIT10ADDRESS

**:TRIGger:EINTerval:EVENT<x>:I2Cbus:CLOCK?**

**Function** Queries all settings related to the clock of the I<sup>2</sup>C bus trigger.

**Syntax** :TRIGger:EINTerval:EVENT<x>:I2Cbus:CLOCK?  
<x> = 1 or 2

**Example** :TRIGGER:EINTERVAL:EVENT1:I2CBUS:CLOCK?  
-> :TRIGGER:EINTERVAL:EVENT1:I2CBUS:CLOCK:SOURCE 1

**:TRIGger:EINTerval:EVENT<x>:I2Cbus:CLOCK:SOURCE**

**Function** Sets the clock trace of the I<sup>2</sup>C bus trigger or queries the current setting.

**Syntax** :TRIGger:EINTerval:EVENT<x>:I2Cbus:CLOCK:SOURCE {<NRf>}  
:TRIGger:EINTerval:EVENT<x>:I2Cbus:CLOCK:SOURCE?  
<x> = 1 or 2  
<NRf> = 1 to 4

**Example** :TRIGGER:EINTERVAL:EVENT1:I2CBUS:CLOCK:SOURCE 1  
:TRIGGER:EINTERVAL:EVENT1:I2CBUS:CLOCK:SOURCE? -> :TRIGGER:EINTERVAL:EVENT1:I2CBUS:CLOCK:SOURCE 1

**:TRIGger:EINTerval:EVENT<x>:I2Cbus:DATA?**

**Function** Queries all settings related to the data of the I<sup>2</sup>C bus trigger.

**Syntax** :TRIGger:EINTerval:EVENT<x>:I2Cbus:DATA?  
<x> = 1 or 2

**Example** :TRIGGER:EINTERVAL:EVENT1:I2CBUS:DATA? -> :TRIGGER:EINTERVAL:EVENT1:I2CBUS:DATA:BYTE 1;CONDITION TRUE;DPOSITION 1;MODE 1;PATTERN1 "10101011";PATTERN2 "10101010";PATTERN3 "10101111";PATTERN4 "10101011";PMODE DONTCARE;SOURCE 1

**:TRIGger:EINTerval:EVENT<x>:I2Cbus:DATA:BYTE**

**Function** Sets the number of data bytes of the I<sup>2</sup>C bus trigger or queries the current setting.

**Syntax** :TRIGger:EINTerval:EVENT<x>:I2Cbus:DATA:BYTE {<NRf>}  
:TRIGger:EINTerval:EVENT<x>:I2Cbus:DATA:BYTE?  
<x> = 1 or 2  
<NRf> = 1 to 4

**Example** :TRIGGER:EINTERVAL:EVENT1:I2CBUS:DATA:BYTE 1  
:TRIGGER:EINTERVAL:EVENT1:I2CBUS:DATA:BYTE? -> :TRIGGER:EINTERVAL:EVENT1:I2CBUS:DATA:BYTE 1

**:TRIGger:EINTerval:EVENT<x>:I2Cbus:DATA:CONDition**

**Function** Sets the determination method (match or not match) of the data of the I<sup>2</sup>C bus trigger or queries the current setting.

**Syntax** :TRIGger:EINTerval:EVENT<x>:I2Cbus:DATA:CONDition {FALSE|TRUE}  
:TRIGger:EINTerval:EVENT<x>:I2Cbus:DATA:CONDition?  
<x> = 1 or 2

**Example** :TRIGGER:EINTERVAL:EVENT1:I2CBUS:DATA:CONDITION TRUE  
:TRIGGER:EINTERVAL:EVENT1:I2CBUS:DATA:CONDITION? -> :TRIGGER:EINTERVAL:EVENT1:I2CBUS:DATA:CONDITION TRUE

**:TRIGger:EINTerval:EVENT<x>:I2Cbus:DATA:DPOSITION**

**Function** Sets the position for comparing the data pattern of the I<sup>2</sup>C bus trigger or queries the current setting.

**Syntax** :TRIGger:EINTerval:EVENT<x>:I2Cbus:DATA:DPOSITION {<NRf>}  
:TRIGger:EINTerval:EVENT<x>:I2Cbus:DATA:DPOSITION?  
<x> = 1 or 2  
<NRf> = 0 to 9999

**Example** :TRIGGER:EINTERVAL:EVENT1:I2CBUS:DATA:DPOSITION 1  
:TRIGGER:EINTERVAL:EVENT1:I2CBUS:DATA:DPOSITION? -> :TRIGGER:EINTERVAL:EVENT1:I2CBUS:DATA:DPOSITION 1

## 5.29 TRIGger Group

### **:TRIGger:EINTerval:EVENT<x>:I2CBus:**

#### **DATA:HEXA<x>**

**Function** Sets the data of the I<sup>2</sup>C bus trigger in hexadecimal notation.

**Syntax** :TRIGger:EINTerval:EVENT<x>:I2CBus:

DATA:HEXA<x> {<String>}

<x> of EVENT<x> = 1 or 2

<x> of HEXA<x> = 1 to 4

<String> = 2 characters by combining '0' to 'F' and 'X'

**Example** :TRIGGER:EINTERVAL:EVENT1:I2CBUS:DATA:  
HEXA1 "AB"

### **:TRIGger:EINTerval:EVENT<x>:I2CBus:**

#### **DATA:MODE**

**Function** Enables/Disables the data conditions of the I<sup>2</sup>C bus trigger or queries the current setting.

**Syntax** :TRIGger:EINTerval:EVENT<x>:I2CBus:

DATA:MODE {<Boolean>}

:TRIGger:EINTerval:EVENT<x>:I2CBus:

DATA:MODE?

<x> = 1 or 2

**Example** :TRIGGER:EINTERVAL:EVENT1:I2CBUS:DATA:  
MODE ON  
:TRIGGER:EINTERVAL:EVENT1:I2CBUS:DATA:  
MODE? -> :TRIGGER:EINTERVAL:EVENT1:  
I2CBUS:DATA:MODE 1

### **:TRIGger:EINTerval:EVENT<x>:I2CBus:**

#### **DATA:PATtern<x>**

**Function** Sets the data of the I<sup>2</sup>C bus trigger in binary notation or queries the current setting.

**Syntax** :TRIGger:EINTerval:EVENT<x>:I2CBus:

DATA:PATtern<x> {<String>}

:TRIGger:EINTerval:EVENT<x>:I2CBus:

DATA:PATtern<x>?

<x> of EVENT<x> = 1 or 2

<x> of PATtern<x> = 1 to 4

<String> = 8 characters by combining '0,' '1,' and 'X'

**Example** :TRIGGER:EINTERVAL:EVENT1:I2CBUS:DATA:  
PATTERN1 "10101011"  
:TRIGGER:EINTERVAL:EVENT1:I2CBUS:DATA:  
PATTERN1? -> :TRIGGER:EINTERVAL:EVENT1:  
I2CBUS:DATA:PATTERN1 "10101011"

### **:TRIGger:EINTerval:EVENT<x>:I2CBus:**

#### **DATA:PMODE**

**Function** Sets the pattern comparison start position mode of the data of the I<sup>2</sup>C bus trigger or queries the current setting.

**Syntax** :TRIGger:EINTerval:EVENT<x>:I2CBus:

DATA:PMODE {DONTcare|SElect}

:TRIGger:EINTerval:EVENT<x>:I2CBus:

DATA:PMODE?

<x> = 1 or 2

**Example** :TRIGGER:EINTERVAL:EVENT1:I2CBUS:DATA:  
PMODE SELECT  
:TRIGGER:EINTERVAL:EVENT1:I2CBUS:DATA:  
PMODE? -> :TRIGGER:EINTERVAL:EVENT1:  
I2CBUS:DATA:PMODE SELECT

### **:TRIGger:EINTerval:EVENT<x>:I2CBus:**

#### **DATA:SOURce**

**Function** Sets the data trace of the I<sup>2</sup>C bus trigger or queries the current setting.

**Syntax** :TRIGger:EINTerval:EVENT<x>:I2CBus:

DATA:SOURce {<Nrf>}

:TRIGger:EINTerval:EVENT<x>:I2CBus:

DATA:SOURce?

<x> = 1 or 2

<Nrf> = 1 to 4

**Example** :TRIGGER:EINTERVAL:EVENT1:I2CBUS:  
DATA:SOURCE 1  
:TRIGGER:EINTERVAL:EVENT1:I2CBUS:  
DATA:SOURCE? -> :TRIGGER:EINTERVAL:  
EVENT1:I2CBUS:DATA:SOURCE 1

### **:TRIGger:EINTerval:EVENT<x>:I2CBus:**

#### **GCALL?**

**Function** Queries all settings related to the general call of the I<sup>2</sup>C bus trigger.

**Syntax** :TRIGger:EINTerval:EVENT<x>:I2CBus:

GCALL?

<x> = 1 or 2

**Example** :TRIGGER:EINTERVAL:EVENT1:I2CBUS:GCALL?  
-> :TRIGGER:EINTERVAL:EVENT1:I2CBUS:  
GCALL:BIT7MADDRESS:PATTERN "10101011";:  
TRIGGER:EINTERVAL:EVENT1:I2CBUS:GCALL:  
SBYTE BIT7MADDRESS



**:TRIGger:EINterval:EVENT<x>:I2Cbus:GCALl:BIT7maddress?**

Function Queries all settings related to the 7-bit master address of the general call of the I<sup>2</sup>C bus trigger.

Syntax :TRIGger:EINterval:EVENT<x>:I2Cbus:GCALl:BIT7maddress?  
<x> = 1 or 2

Example :TRIGGER:EINTERVAL:EVENT1:I2CBUS:GCALL:BIT7MADDRESS? -> :TRIGGER:EINTERVAL:EVENT1:I2CBUS:GCALL:BIT7MADDRESS:PATTERN "1010101"

**:TRIGger:EINterval:EVENT<x>:I2Cbus:GCALl:BIT7maddress:HEXA**

Function Sets the 7-bit master address of the general call of the I<sup>2</sup>C bus trigger in hexadecimal notation.

Syntax :TRIGger:EINterval:EVENT<x>:I2Cbus:GCALl:BIT7maddress:HEXA {<String>}  
<x> = 1 or 2  
<String> = 2 characters by combining '0' to 'F' and 'X' (bit 0 is fixed 1)

Example :TRIGGER:EINTERVAL:EVENT1:I2CBUS:GCALL:BIT7MADDRESS:HEXA "AB"

**:TRIGger:EINterval:EVENT<x>:I2Cbus:GCALl:BIT7maddress:PATtern**

Function Sets the 7-bit master address of the general call of the I<sup>2</sup>C bus trigger in binary notation or queries the current setting.

Syntax :TRIGger:EINterval:EVENT<x>:I2Cbus:GCALl:BIT7maddress:PATtern {<String>}  
:TRIGger:EINterval:EVENT<x>:I2Cbus:GCALl:BIT7maddress:PATtern?  
<x> = 1 or 2  
<String> = 7 characters by combining '0','1,' and 'X'

Example :TRIGGER:EINTERVAL:EVENT1:I2CBUS:GCALL:BIT7MADDRESS:PATTERN "1010101"  
:TRIGGER:EINTERVAL:EVENT1:I2CBUS:GCALL:BIT7MADDRESS:PATTERN? -> :TRIGGER:EINTERVAL:EVENT1:I2CBUS:GCALL:BIT7MADDRESS:PATTERN "1010101"

**:TRIGger:EINterval:EVENT<x>:I2Cbus:GCALl:SBYte (Second Byte)**

Function Sets the second byte type of the general call of the I<sup>2</sup>C bus trigger or queries the current setting.

Syntax :TRIGger:EINterval:EVENT<x>:I2Cbus:GCALl:SBYte {BIT7maddress|DONTcare|H04|H06}  
:TRIGger:EINterval:EVENT<x>:I2Cbus:GCALl:SBYte?  
<x> = 1 or 2

Example :TRIGGER:EINTERVAL:EVENT1:I2CBUS:GCALL:SBYTE BIT7MADDRESS  
:TRIGGER:EINTERVAL:EVENT1:I2CBUS:GCALL:SBYTE? -> :TRIGGER:EINTERVAL:EVENT1:I2CBUS:GCALL:SBYTE BIT7MADDRESS

**:TRIGger:EINterval:EVENT<x>:I2Cbus:MODE**

Function Sets the trigger mode of the I<sup>2</sup>C bus trigger or queries the current setting.

Syntax :TRIGger:EINterval:EVENT<x>:I2Cbus:MODE {ADATa|ESTart|GCALl|NAIgnore|SBHSmode}  
:TRIGger:EINterval:EVENT<x>:I2Cbus:MODE?  
<x> = 1 or 2

Example :TRIGGER:EINTERVAL:EVENT1:I2CBUS:MODE ADATA  
:TRIGGER:EINTERVAL:EVENT1:I2CBUS:MODE? -> :TRIGGER:EINTERVAL:EVENT1:I2CBUS:MODE ADATA

**:TRIGger:EINterval:EVENT<x>:I2Cbus:NAIgnore?**

Function Queries all settings related to the NON ACK ignore mode of the I<sup>2</sup>C bus trigger.

Syntax :TRIGger:EINterval:EVENT<x>:I2Cbus:NAIgnore?  
<x> = 1 or 2

Example :TRIGGER:EINTERVAL:EVENT1:I2CBUS:NAIGNORE? -> :TRIGGER:EINTERVAL:EVENT1:I2CBUS:NAIGNORE:HSMODE 1;RACCESS 1;SBYTE 1

## 5.29 TRIGger Group

### **:TRIGger:EINTerval:EVENT<x>:I2Cbus:NAIGNore:HSMode**

**Function** Sets whether to ignore NON ACK in high speed mode of the I<sup>2</sup>C bus trigger or queries the current setting.

**Syntax** :TRIGger:EINTerval:EVENT<x>:I2Cbus:NAIGNore:HSMode {<Boolean>}  
:TRIGger:EINTerval:EVENT<x>:I2Cbus:NAIGNore:HSMode?  
<x> = 1 or 2

**Example** :TRIGGER:EINTERVAL:EVENT1:I2CBUS:  
NAIGNORE:HSMODE ON  
:TRIGGER:EINTERVAL:EVENT1:I2CBUS:  
NAIGNORE:HSMODE? -> :TRIGGER:EINTERVAL:  
EVENT1:I2CBUS:NAIGNORE:HSMODE 1

### **:TRIGger:EINTerval:EVENT<x>:I2Cbus:NAIGNore:RACcEss**

**Function** Sets whether to ignore NON ACK in read access mode of the I<sup>2</sup>C bus trigger or queries the current setting.

**Syntax** :TRIGger:EINTerval:EVENT<x>:I2Cbus:NAIGNore:RACcEss {<Boolean>}  
:TRIGger:EINTerval:EVENT<x>:I2Cbus:NAIGNore:RACcEss?  
<x> = 1 or 2

**Example** :TRIGGER:EINTERVAL:EVENT1:I2CBUS:  
NAIGNORE:RACCESS ON  
:TRIGGER:EINTERVAL:EVENT1:I2CBUS:  
NAIGNORE:RACCESS? -> :TRIGGER:EINTERVAL:  
EVENT1:I2CBUS:NAIGNORE:RACCESS 1

### **:TRIGger:EINTerval:EVENT<x>:I2Cbus:NAIGNore:SBYTe (Start Byte)**

**Function** Sets whether to ignore NON ACK in the start byte of the I<sup>2</sup>C bus trigger or queries the current setting.

**Syntax** :TRIGger:EINTerval:EVENT<x>:I2Cbus:NAIGNore:SBYTe {<Boolean>}  
:TRIGger:EINTerval:EVENT<x>:I2Cbus:NAIGNore:SBYTe?  
<x> = 1 or 2

**Example** :TRIGGER:EINTERVAL:EVENT1:I2CBUS:  
NAIGNORE:SBYTE ON  
:TRIGGER:EINTERVAL:EVENT1:I2CBUS:  
NAIGNORE:SBYTE? -> :TRIGGER:EINTERVAL:  
EVENT1:I2CBUS:NAIGNORE:SBYTE 1

### **:TRIGger:EINTerval:EVENT<x>:I2Cbus:SBHSMode?**

**Function** Queries all settings related to the start byte and high speed mode of the I<sup>2</sup>C bus trigger.

**Syntax** :TRIGger:EINTerval:EVENT<x>:I2Cbus:SBHSMode?  
<x> = 1 or 2

**Example** :TRIGGER:EINTERVAL:EVENT1:I2CBUS:  
SBHSMODE? -> :TRIGGER:EINTERVAL:EVENT1:  
I2CBUS:SBHSMODE:TYPE HSMODE

### **:TRIGger:EINTerval:EVENT<x>:I2Cbus:SBHSMode:TYPE**

**Function** Sets the type of the start byte or high speed mode of the I<sup>2</sup>C bus trigger or queries the current setting.

**Syntax** :TRIGger:EINTerval:EVENT<x>:I2Cbus:SBHSMode:TYPE {HSMODE|SBYTE}  
:TRIGger:EINTerval:EVENT<x>:I2Cbus:SBHSMode:TYPE?  
<x> = 1 or 2

**Example** :TRIGGER:EINTERVAL:EVENT1:I2CBUS:  
SBHSMODE:TYPE HSMODE  
:TRIGGER:EINTERVAL:EVENT1:I2CBUS:  
SBHSMODE:TYPE? -> :TRIGGER:EINTERVAL:  
EVENT1:I2CBUS:SBHSMODE:TYPE HSMODE

### **:TRIGger:EINTerval:EVENT<x>:LINBus?**

**Function** Queries all settings related to LIN bus signal triggers of each event.

**Syntax** :TRIGger:EINTerval:EVENT<x>:LINBus?  
<x> = 1 or 2

**Example** :TRIGGER:EINTERVAL:EVENT1:LINBUS?  
-> :TRIGGER:EINTERVAL:EVENT1:LINBUS:  
BRATE 19200;SOURCE 1

### **:TRIGger:EINTerval:EVENT<x>:LINBus:BRATe**

**Function** Sets the LIN bus signal trigger bitrate (data transfer rate) or queries the current setting.

**Syntax** :TRIGger:EINTerval:EVENT<x>:LINBus:BRATe {<Nrf>|USER,<Nrf>}  
:TRIGger:EINTerval:EVENT<x>:LINBus:BRATe?  
<x> = 1 or 2

<Nrf> = 1200, 2400, 4800, 9600, 19200  
<Nrf> for USER = See the main unit User's Manual.  
**Example** :TRIGGER:EINTERVAL:EVENT1:LINBUS:  
BRATE 19200  
:TRIGGER:EINTERVAL:EVENT1:LINBUS:BRATE?  
-> :TRIGGER:EINTERVAL:EVENT1:LINBUS:  
BRATE 19200

**:TRIGger:EINTerval:EVENT<x>:LINBus:SOURCE**

**Function** Sets the LIN bus signal trigger source or queries the current setting.

**Syntax** :TRIGger:EINTerval:EVENT<x>:LINBus:SOURCE {<NRf>}  
:TRIGger:EINTerval:EVENT<x>:LINBus:SOURCE?  
<x> = 1 or 2  
<NRf> = 1 to 4

**Example** :TRIGGER:EINTERVAL:EVENT1:LINBUS:SOURCE 1  
:TRIGGER:EINTERVAL:EVENT1:LINBUS:SOURCE? -> :TRIGGER:EINTERVAL:EVENT1:LINBUS:SOURCE 1

**:TRIGger:EINTerval:EVENT<x>:SPATtern? (Serial Pattern)**

**Function** Queries all settings related to the serial pattern trigger of the event.

**Syntax** :TRIGger:EINTerval:EVENT<x>:SPATtern?  
<x> = 1 or 2

**Example** :TRIGGER:EINTERVAL:EVENT1:SPATTERN?  
-> :TRIGGER:EINTERVAL:EVENT1:SPATTERN:BITRATE 1.000E+00;CLOCK:MODE 1;POLARITY FALL;SOURCE 1;:TRIGGER:EINTERVAL:EVENT1:SPATTERN:CS 1;DATA:ACTIVE HIGH;SOURCE 1;:TRIGGER:EINTERVAL:EVENT1:SPATTERN:LATCH:SOURCE 1;POLARITY FALL;:TRIGGER:EINTERVAL:EVENT1:SPATTERN:PATTERN "1100110111101111"

**:TRIGger:EINTerval:EVENT<x>:SPATtern:BITRate**

**Function** Sets the bit rate of the serial pattern trigger or queries the current setting.

**Syntax** :TRIGger:EINTerval:EVENT<x>:SPATtern:BITRate {<NRf>}  
:TRIGger:EINTerval:EVENT<x>:SPATtern:BITRate?  
<x> = 1 or 2  
<NRf> = 1 to 50M (bps)

**Example** :TRIGGER:EINTERVAL:EVENT1:SPATTERN:BITRATE 1  
:TRIGGER:EINTERVAL:EVENT1:SPATTERN:BITRATE? -> :TRIGGER:EINTERVAL:EVENT1:SPATTERN:BITRATE 1.000E+00

**Description** This command is valid when :TRIGger:EINTerval:EVENT<x>:SPATtern:CLOCK:MODE OFF.

**:TRIGger:EINTerval:EVENT<x>:SPATtern:CLEAr**

**Function** Clears the entire pattern of the serial pattern trigger (to don't care).

**Syntax** :TRIGger:EINTerval:EVENT<x>:SPATtern:CLEAr  
<x> = 1 or 2

**Example** :TRIGGER:EINTERVAL:EVENT1:SPATTERN:CLEAr

**:TRIGger:EINTerval:EVENT<x>:SPATtern:CLOCK?**

**Function** Queries all settings related to clock of the serial pattern trigger.

**Syntax** :TRIGger:EINTerval:EVENT<x>:SPATtern:CLOCK?  
<x> = 1 or 2

**Example** :TRIGGER:EINTERVAL:EVENT1:SPATTERN:CLOCK? -> :TRIGGER:EINTERVAL:EVENT1:SPATTERN:CLOCK:MODE 1;POLARITY FALL;SOURCE 1

**:TRIGger:EINTerval:EVENT<x>:SPATtern:CLOCK:MODE**

**Function** Enables/Disables the clock of the serial pattern trigger or queries the current setting.

**Syntax** :TRIGger:EINTerval:EVENT<x>:SPATtern:CLOCK:MODE {<Boolean>}  
:TRIGger:EINTerval:EVENT<x>:SPATtern:CLOCK:MODE?  
<x> = 1 or 2

**Example** :TRIGGER:EINTERVAL:EVENT1:SPATTERN:CLOCK:MODE ON  
:TRIGGER:EINTERVAL:EVENT1:SPATTERN:CLOCK:MODE? -> :TRIGGER:EINTERVAL:EVENT1:SPATTERN:CLOCK:MODE 1

**:TRIGger:EINTerval:EVENT<x>:SPATtern:CLOCK:POLarity**

**Function** Sets the polarity of the clock trace of the serial pattern trigger or queries the current setting.

**Syntax** :TRIGger:EINTerval:EVENT<x>:SPATtern:CLOCK:POLarity {FALL|RISE}  
:TRIGger:EINTerval:EVENT<x>:SPATtern:CLOCK:POLarity?  
<x> = 1 or 2

**Example** :TRIGGER:EINTERVAL:EVENT1:SPATTERN:CLOCK:POLARITY FALL  
:TRIGGER:EINTERVAL:EVENT1:SPATTERN:CLOCK:POLARITY? -> :TRIGGER:EINTERVAL:EVENT1:SPATTERN:CLOCK:POLARITY FALL

**Description** This command is valid when :TRIGger:EINTerval:EVENT<x>:SPATtern:CLOCK:MODE ON.

## 5.29 TRIGger Group

### **:TRIGger:EINTerval:EVENT<x>:SPATtern:CLOCK:SOURce**

**Function** Sets the clock trace of the serial pattern trigger or queries the current setting.

**Syntax** :TRIGger:EINTerval:EVENT<x>:SPATtern:  
CLOCK:SOURce {<NRf>}  
:TRIGger:EINTerval:EVENT<x>:SPATtern:  
CLOCK:SOURce?  
<x> = 1 or 2  
<NRf> = 1 to 4

**Example** :TRIGGER:EINTERVAL:EVENT1:SPATTERN:  
CLOCK:SOURCE 1  
:TRIGGER:EINTERVAL:EVENT1:SPATTERN:  
CLOCK:SOURCE? -> :TRIGGER:EINTERVAL:  
EVENT1:SPATTERN:CLOCK:SOURCE 1

**Description** This command is valid when :TRIGger:  
EINTerval:EVENT<x>:SPATtern:CLOCK:  
MODE ON.

### **:TRIGger:EINTerval:EVENT<x>:SPATtern:CS**

**Function** Enables/Disables the chip select of the serial pattern trigger or queries the current setting.

**Syntax** :TRIGger:EINTerval:EVENT<x>:SPATtern:  
CS {<Boolean>}  
:TRIGger:EINTerval:EVENT<x>:SPATtern:  
CS?  
<x> = 1 or 2

**Example** :TRIGGER:EINTERVAL:EVENT1:SPATTERN:  
CS ON  
:TRIGGER:EINTERVAL:EVENT1:SPATTERN:CS?  
-> :TRIGGER:EINTERVAL:EVENT1:SPATTERN:  
CS 1

**Description** This command is valid when :TRIGger:  
EINTerval:EVENT<x>:SPATtern:CLOCK:  
MODE ON.

### **:TRIGger:EINTerval:EVENT<x>:SPATtern:DATA?**

**Function** Queries all settings related to data of the serial pattern trigger.

**Syntax** :TRIGger:EINTerval:EVENT<x>:SPATtern:  
DATA?  
<x> = 1 or 2

**Example** :TRIGGER:EINTERVAL:EVENT1:SPATTERN:  
DATA? -> :TRIGGER:EINTERVAL:EVENT1:  
SPATTERN:DATA:ACTIVE HIGH;SOURCE 1

### **:TRIGger:EINTerval:EVENT<x>:SPATtern:DATA:ACTive**

**Function** Sets the active level of the data of the serial pattern trigger or queries the current setting.

**Syntax** :TRIGger:EINTerval:EVENT<x>:SPATtern:  
DATA:ACTive {HIGH|LOW}  
:TRIGger:EINTerval:EVENT<x>:SPATtern:  
DATA:ACTive?  
<x> = 1 or 2

**Example** :TRIGGER:EINTERVAL:EVENT1:SPATTERN:  
DATA:ACTIVE HIGH  
:TRIGGER:EINTERVAL:EVENT1:SPATTERN:  
DATA:ACTIVE? -> :TRIGGER:EINTERVAL:  
EVENT1:SPATTERN:DATA:ACTIVE HIGH

### **:TRIGger:EINTerval:EVENT<x>:SPATtern:DATA:SOURce**

**Function** Sets the data trace of the serial pattern trigger or queries the current setting.

**Syntax** :TRIGger:EINTerval:EVENT<x>:SPATtern:  
DATA:SOURce {<NRf>}  
:TRIGger:EINTerval:EVENT<x>:SPATtern:  
DATA:SOURce?  
<x> = 1 or 2  
<NRf> = 1 to 4

**Example** :TRIGGER:EINTERVAL:EVENT1:SPATTERN:  
DATA:SOURCE 1  
:TRIGGER:EINTERVAL:EVENT1:SPATTERN:  
DATA:SOURCE? -> :TRIGGER:EINTERVAL:  
EVENT1:SPATTERN:DATA:SOURCE 1

### **:TRIGger:EINTerval:EVENT<x>:SPATtern:HEXA**

**Function** Sets the pattern of the serial pattern trigger in hexadecimal notation.

**Syntax** :TRIGger:EINTerval:EVENT<x>:SPATtern:  
HEXA {<String>}  
<x> = 1 or 2  
<String> = Up to 32 characters by combining '0' to 'F'  
and 'X'

**Example** :TRIGGER:EINTERVAL:EVENT1:SPATTERN:  
HEXA "ABCD"

### **:TRIGger:EINTerval:EVENT<x>:SPATtern:LATCh?**

**Function** Queries all settings related to latch of the serial pattern trigger.

**Syntax** :TRIGger:EINTerval:EVENT<x>:SPATtern:  
LATCh?  
<x> = 1 or 2

**Example** :TRIGGER:EINTERVAL:EVENT1:SPATTERN:  
LATCH? -> :TRIGGER:EINTERVAL:EVENT1:  
SPATTERN:LATCH:SOURCE 1;POLARITY FALL

**:TRIGger:EINterval:EVENT<x>:SPATtern:  
LATCh:POLarity**

**Function** Sets the polarity of the latch trace of the serial pattern trigger or queries the current setting.

**Syntax** :TRIGger:EINterval:EVENT<x>:SPATtern:  
LATCh:POLarity {FALL|RISE}  
:TRIGger:EINterval:EVENT<x>:SPATtern:  
LATCh:POLarity?  
<x> = 1 or 2

**Example** :TRIGGER:EINTERVAL:EVENT1:SPATTERN:  
LATCH:POLARITY FALL  
:TRIGGER:EINTERVAL:EVENT1:SPATTERN:  
LATCH:POLARITY? -> :TRIGGER:EINTERVAL:  
EVENT1:SPATTERN:LATCH:POLARITY FALL

**Description** • This command is valid when :TRIGger:  
EINterval:EVENT<x>:SPATtern:  
CLOCK:MODE ON.  
• This command is invalid if :TRIGger:  
EINterval:EVENT<x>:SPATtern:  
LATCH:SOURCE NONE.

**:TRIGger:EINterval:EVENT<x>:SPATtern:  
LATCh:SOURce**

**Function** Sets the latch trace of the serial pattern trigger or queries the current setting.

**Syntax** :TRIGger:EINterval:EVENT<x>:SPATtern:  
LATCh:SOURce {<NRf>|NONE}  
:TRIGger:EINterval:EVENT<x>:SPATtern:  
LATCh:SOURce?  
<x> = 1 or 2  
<NRf> = 1 to 4

**Example** :TRIGGER:EINTERVAL:EVENT1:SPATTERN:  
LATCH:SOURCE 1  
:TRIGGER:EINTERVAL:EVENT1:SPATTERN:  
LATCH:SOURCE? -> :TRIGGER:EINTERVAL:  
EVENT1:SPATTERN:LATCH:SOURCE 1

**Description** This command is valid when :TRIGger:  
EINterval:EVENT<x>:SPATtern:CLOCK:  
MODE ON.

**:TRIGger:EINterval:EVENT<x>:SPATtern:  
PATtern**

**Function** Sets the pattern of the serial pattern trigger in binary notation or queries the current setting.

**Syntax** :TRIGger:EINterval:EVENT<x>:SPATtern:  
PATtern {<String>}  
:TRIGger:EINterval:EVENT<x>:SPATtern:  
PATtern?  
<x> = 1 or 2  
<String> = Up to 128 characters by combining '0', '1',  
and 'X'

**Example** :TRIGGER:EINTERVAL:EVENT1:SPATTERN:  
PATTERN "1100110111101111"  
:TRIGGER:EINTERVAL:EVENT1:SPATTERN:  
PATTERN? -> :TRIGGER:EINTERVAL:EVENT1:  
SPATTERN:PATTERN "1100110111101111"

**:TRIGger:EINterval:EVENT<x>:SPIBUS?**

**Function** Queries all settings related to the SPI bus trigger of the event.

**Syntax** :TRIGger:EINterval:EVENT<x>:SPIBUS?  
<x> = 1 or 2

**Example** :TRIGGER:EINTERVAL:EVENT1:SPIBUS?  
-> :TRIGGER:EINTERVAL:EVENT1:SPIBUS:  
BITORDER LSBFIRST;CLOCK:POLARITY FALL;  
SOURCE 1;:TRIGGER:EINTERVAL:EVENT1:  
SPIBUS:CS:ACTIVE HIGH;SOURCE 1;:  
TRIGGER:EINTERVAL:EVENT1:SPIBUS:DATA1:  
BYTE 1;CONDITION TRUE;DPOSITION 1;  
PATTERN1 "00010010";  
PATTERN2 "00110100";  
PATTERN3 "01010110";  
PATTERN4 "00010010";SOURCE 3;:TRIGGER:  
EINTERVAL:EVENT1:SPIBUS:DATA2:BYTE 4;  
CONDITION TRUE;DPOSITION 1;  
PATTERN1 "00010010";  
PATTERN2 "00110100";  
PATTERN3 "01010110";  
PATTERN4 "00010010";SOURCE 3;:TRIGGER:  
EINTERVAL:EVENT1:SPIBUS:MODE WIRE3

**:TRIGger:EINterval:EVENT<x>:SPIBUS:  
BITorder**

**Function** Sets the bit order of the SPI bus trigger or queries the current setting.

**Syntax** :TRIGger:EINterval:EVENT<x>:SPIBUS:  
BITorder {LSBFirst|MSBFirst}  
:TRIGger:EINterval:EVENT<x>:SPIBUS:  
BITorder?  
<x> = 1 or 2

**Example** :TRIGGER:EINTERVAL:EVENT1:SPIBUS:  
BITORDER LSBFIRST  
:TRIGGER:EINTERVAL:EVENT1:SPIBUS:  
BITORDER? -> :TRIGGER:EINTERVAL:  
EVENT1:SPIBUS:BITORDER LSBFIRST

## 5.29 TRIGger Group

### **:TRIGger:EINterval:EVENT<x>:SPIBus:CLOCK?**

**Function** Queries all settings related to the clock of the SPI bus trigger.

**Syntax** :TRIGger:EINterval:EVENT<x>:SPIBus:CLOCK?  
<x> = 1 or 2

**Example** :TRIGGER:EINTERVAL:EVENT1:SPIBUS:CLOCK?  
-> :TRIGGER:EINTERVAL:EVENT1:SPIBUS:CLOCK:POLARITY FALL;SOURCE 1

### **:TRIGger:EINterval:EVENT<x>:SPIBus:CLOCK:POLarity**

**Function** Sets the polarity of the clock trace of the SPI bus trigger or queries the current setting.

**Syntax** :TRIGger:EINterval:EVENT<x>:SPIBus:CLOCK:POLarity {FALL|RISE}  
:TRIGger:EINterval:EVENT<x>:SPIBus:CLOCK:POLarity?  
<x> = 1 or 2

**Example** :TRIGGER:EINTERVAL:EVENT1:SPIBUS:CLOCK:POLARITY FALL  
:TRIGGER:EINTERVAL:EVENT1:SPIBUS:CLOCK:POLARITY? -> :TRIGGER:EINTERVAL:EVENT1:SPIBUS:CLOCK:POLARITY FALL

### **:TRIGger:EINterval:EVENT<x>:SPIBus:CLOCK:SOURce**

**Function** Sets the clock trace of the SPI bus trigger or queries the current setting.

**Syntax** :TRIGger:EINterval:EVENT<x>:SPIBus:CLOCK:SOURce {<NRf>}  
:TRIGger:EINterval:EVENT<x>:SPIBus:CLOCK:SOURce?  
<x> = 1 or 2  
<NRf> = 1 to 4

**Example** :TRIGGER:EINTERVAL:EVENT1:SPIBUS:CLOCK:SOURCE 1  
:TRIGGER:EINTERVAL:EVENT1:SPIBUS:CLOCK:SOURCE? -> :TRIGGER:EINTERVAL:EVENT1:SPIBUS:CLOCK:SOURCE 1

### **:TRIGger:EINterval:EVENT<x>:SPIBus:CS?**

**Function** Queries all settings related to the chip select of the SPI bus trigger.

**Syntax** :TRIGger:EINterval:EVENT<x>:SPIBus:CS?  
<x> = 1 or 2

**Example** :TRIGGER:EINTERVAL:EVENT1:SPIBUS:CS?  
-> :TRIGGER:EINTERVAL:EVENT1:SPIBUS:CS:ACTIVE HIGH;SOURCE 1

### **:TRIGger:EINterval:EVENT<x>:SPIBus:CS:ACTIVE**

**Function** Sets the active level of the chip select of the SPI bus trigger or queries the current setting.

**Syntax** :TRIGger:EINterval:EVENT<x>:SPIBus:CS:ACTIVE {HIGH|LOW}  
:TRIGger:EINterval:EVENT<x>:SPIBus:CS:ACTIVE?  
<x> = 1 or 2

**Example** :TRIGGER:EINTERVAL:EVENT1:SPIBUS:CS:ACTIVE HIGH  
:TRIGGER:EINTERVAL:EVENT1:SPIBUS:CS:ACTIVE? -> :TRIGGER:EINTERVAL:EVENT1:SPIBUS:CS:ACTIVE HIGH

### **:TRIGger:EINterval:EVENT<x>:SPIBus:CS:SOURce**

**Function** Sets the chip select trace of the SPI bus trigger or queries the current setting.

**Syntax** :TRIGger:EINterval:EVENT<x>:SPIBus:CS:SOURce {<NRf>}  
:TRIGger:EINterval:EVENT<x>:SPIBus:CS:SOURce?  
<x> = 1 or 2  
<NRf> = 1 to 4

**Example** :TRIGGER:EINTERVAL:EVENT1:SPIBUS:CS:SOURCE 1  
:TRIGGER:EINTERVAL:EVENT1:SPIBUS:CS:SOURCE? -> :TRIGGER:EINTERVAL:EVENT1:SPIBUS:CS:SOURCE 1

### **:TRIGger:EINterval:EVENT<x>:SPIBus:DATA<x>?**

**Function** Queries all settings related to the data of the SPI bus trigger.

**Syntax** :TRIGger:EINterval:EVENT<x>:SPIBus:DATA<x>?  
<x> of EVENT<x> = 1 or 2  
<x> of DATA<x> = 1 or 2

**Example** :TRIGGER:EINTERVAL:EVENT1:SPIBUS:DATA1?  
-> :TRIGGER:EINTERVAL:EVENT1:SPIBUS:DATA1:BYTE 1;CONDITION TRUE;  
DPOSITION 1;PATTERN1 "00010010";  
PATTERN2 "00110100";  
PATTERN3 "01010110";  
PATTERN4 "00010010";SOURCE 3

**Description** DATA2 is valid when :TRIGger:EINterval:EVENT<x>:SPIBus:MODE WIRE4.

**:TRIGger:EINterval:EVENT<x>:SPIBus:DATA<x>:BYTE**

**Function** Sets the number of bytes of the data of the SPI bus trigger or queries the current setting.

**Syntax**  
 :TRIGger:EINterval:EVENT<x>:SPIBus:  
 DATA<x>:BYTE {<NRF>}  
 :TRIGger:EINterval:EVENT<x>:SPIBus:  
 DATA<x>:BYTE?  
 <x> of EVENT<x> = 1 or 2  
 <x> of DATA<x> = 1 or 2  
 <NRF> = 1 to 4

**Example**  
 :TRIGGER:EINTERVAL:EVENT1:SPIBUS:DATA1:  
 BYTE 1  
 :TRIGGER:EINTERVAL:EVENT1:SPIBUS:DATA1:  
 BYTE? -> :TRIGGER:EINTERVAL:EVENT1:  
 SPIBUS:DATA1:BYTE 1

**:TRIGger:EINterval:EVENT<x>:SPIBus:DATA<x>:CONDition**

**Function** Sets the determination method (match or not match) of the data of the SPI bus trigger or queries the current setting.

**Syntax**  
 :TRIGger:EINterval:EVENT<x>:SPIBus:  
 DATA<x>:CONDition {FALSE|TRUE}  
 :TRIGger:EINterval:EVENT<x>:SPIBus:  
 DATA<x>:CONDition?  
 <x> of EVENT<x> = 1 or 2  
 <x> of DATA<x> = 1 or 2

**Example**  
 :TRIGGER:EINTERVAL:EVENT1:SPIBUS:DATA1:  
 CONDITION TRUE  
 :TRIGGER:EINTERVAL:EVENT1:SPIBUS:DATA1:  
 CONDITION? -> :TRIGGER:EINTERVAL:  
 EVENT1:SPIBUS:DATA1:CONDITION TRUE

**:TRIGger:EINterval:EVENT<x>:SPIBus:DATA<x>:DPOSITioN**

**Function** Sets the pattern comparison start position of the data of the SPI bus trigger or queries the current setting.

**Syntax**  
 :TRIGger:EINterval:EVENT<x>:SPIBus:  
 DATA<x>:DPOSITioN {<NRF>}  
 :TRIGger:EINterval:EVENT<x>:SPIBus:  
 DATA<x>:DPOSITioN?  
 <x> of EVENT<x> = 1 or 2  
 <x> of DATA<x> = 1 or 2  
 <NRF> = 0 to 9999

**Example**  
 :TRIGGER:EINTERVAL:EVENT1:SPIBUS:DATA1:  
 DPOSITION 1  
 :TRIGGER:EINTERVAL:EVENT1:SPIBUS:DATA1:  
 DPOSITION? -> :TRIGGER:EINTERVAL:  
 EVENT1:SPIBUS:DATA1:DPOSITION 1

**:TRIGger:EINterval:EVENT<x>:SPIBus:DATA<x>:HEXA<x>**

**Function** Sets the data of the SPI bus trigger in hexadecimal notation.

**Syntax**  
 :TRIGger:EINterval:EVENT<x>:SPIBus:  
 DATA<x>:HEXA<x> {<String>}  
 <x> of EVENT<x> = 1 or 2  
 <x> of DATA<x> = 1 or 2  
 <x> of HEXA<x> = 1 to 4  
 <String> = 2 characters by combining '0' to 'F' and 'X'

**Example**  
 :TRIGGER:EINTERVAL:EVENT1:SPIBUS:DATA1:  
 HEXA1 "AB"

**:TRIGger:EINterval:EVENT<x>:SPIBus:DATA<x>:PATtern<x>**

**Function** Sets the data of the SPI bus trigger in binary notation or queries the current setting.

**Syntax**  
 :TRIGger:EINterval:EVENT<x>:SPIBus:  
 DATA<x>:PATtern<x> {<String>}  
 :TRIGger:EINterval:EVENT<x>:  
 SPIBus:  
 DATA<x>:PATtern<x>?  
 <x> of EVENT<x> = 1 or 2  
 <x> of DATA<x> = 1 or 2  
 <x> of PATtern<x> = 1 to 4  
 <String> = 8 characters by combining '0', '1,' and 'X'

**Example**  
 :TRIGGER:EINTERVAL:EVENT1:SPIBUS:DATA1:  
 PATTERN1 "10101011"  
 :TRIGGER:EINTERVAL:EVENT1:SPIBUS:DATA1:  
 PATTERN1? -> :TRIGGER:EINTERVAL:EVENT1:  
 SPIBUS:DATA1:PATTERN1 "10101011"

**:TRIGger:EINterval:EVENT<x>:SPIBus:DATA<x>:SOURce**

**Function** Sets the trace of the data of the SPI bus trigger or queries the current setting.

**Syntax**  
 :TRIGger:EINterval:EVENT<x>:SPIBus:  
 DATA<x>:SOURce {<NRF>}  
 :TRIGger:EINterval:EVENT<x>:SPIBus:  
 DATA<x>:SOURce?  
 <x> of EVENT<x> = 1 or 2  
 <x> of DATA<x> = 1 or 2  
 <NRF> = 1 to 4

**Example**  
 :TRIGGER:EINTERVAL:EVENT1:SPIBUS:DATA1:  
 SOURCE 1  
 :TRIGGER:EINTERVAL:EVENT1:SPIBUS:DATA1:  
 SOURCE? -> :TRIGGER:EINTERVAL:EVENT1:  
 SPIBUS:DATA1:SOURCE 1

## 5.29 TRIGger Group

### **:TRIGger:EINterval:EVENT<x>:SPIBus:MODE**

**Function** Sets the wiring system of the SPI bus trigger (three-wire or four-wire) or queries the current setting.

**Syntax** :TRIGger:EINterval:EVENT<x>:SPIBus:  
MODE {WIRE3|WIRE4}  
:TRIGger:EINterval:EVENT<x>:SPIBus:  
MODE?  
<x> = 1 or 2

**Example** :TRIGGER:EINTERVAL:EVENT1:SPIBUS:  
MODE WIRE3  
:TRIGGER:EINTERVAL:EVENT1:SPIBUS:MODE?  
-> :TRIGGER:EINTERVAL:EVENT1:SPIBUS:  
MODE WIRE3

### **:TRIGger:EINterval:EVENT<x>:STATE?**

**Function** Queries all settings related to condition to be satisfied of the event.

**Syntax** :TRIGger:EINterval:EVENT<x>:STATE?  
<x> = 1 or 2

**Example** :TRIGGER:EINTERVAL:EVENT1:STATE? -> :  
TRIGGER:EINTERVAL:EVENT1:STATE:  
CHANNEL1 DONTCARE;CHANNEL2 DONTCARE;  
CHANNEL3 DONTCARE;CHANNEL4 DONTCARE;  
LOGIC AND

### **:TRIGger:EINterval:EVENT<x>:STATE:CHANNEL<x>**

**Function** Sets the condition to be satisfied of the channel or queries the current setting.

**Syntax** :TRIGger:EINterval:EVENT<x>:STATE:  
CHANNEL<x> {DONTcare|HIGH|IN|LOW|OUT}  
:TRIGger:EINterval:EVENT<x>:STATE:  
CHANNEL<x>?  
<x> of EVENT<x> = 1 or 2  
<x> of CHANNEL<x> = 1 to 4

**Example** :TRIGGER:EINTERVAL:EVENT1:STATE:  
CHANNEL1 HIGH  
:TRIGGER:EINTERVAL:EVENT1:STATE:  
CHANNEL1? -> :TRIGGER:EINTERVAL:EVENT1:  
STATE:CHANNEL1 HIGH

**Description** • This command is valid when :TRIGger:  
EINterval:EVENT<x>:TYPE EQUalify|I2CB  
us|PQUalify|PStAte|SPATtern|StAte.  
• {HIGH|LOW} is valid when :TRIGger:  
EINterval:EVENT<x>:TYPE I2CBus|  
SPATtern.  
• {IN|OUT} is valid when :TRIGger:EINterval:  
EVENT<x>:TYPE EQUalify|PQUalify|PStA  
te|StAte and TRIGger:SOURce:CHANnel<x>:  
WINDow ON. {HIGH|LOW} is valid when :  
TRIGger:EINterval:EVENT<x>:TYPE EQUal  
ify|PQUalify|PStAte|StAte and TRIGger:  
SOURce:CHANnel<x>:WINDow OFF.

### **:TRIGger:EINterval:EVENT<x>:STATE:LOGic**

**Function** Sets the logic of the condition to be satisfied or queries the current setting.

**Syntax** :TRIGger:EINterval:EVENT<x>:STATE:  
LOGic {AND|OR}  
:TRIGger:EINterval:EVENT<x>:STATE:  
LOGic?  
<x> = 1 or 2

**Example** :TRIGGER:EINTERVAL:EVENT1:STATE:  
LOGIC AND  
:TRIGGER:EINTERVAL:EVENT1:STATE:LOGIC?  
-> :TRIGGER:EINTERVAL:EVENT1:STATE:  
LOGIC AND

**Description** This command is valid when :TRIGger:  
EINterval:EVENT<x>:TYPE EQUalify|  
I2CBus|PQUalify|PStAte|SPATtern|StAte.

### **:TRIGger:EINterval:EVENT<x>:TYPE**

**Function** Sets the trigger type of the event or queries the current setting.

**Syntax** :TRIGger:EINterval:EVENT<x>:  
TYPE {CANBus|EDGE|EQUalify|I2CBus|  
LINBus|PQUalify|PStAte|PULSe|SPATtern|  
SPIBus|StAte}  
:TRIGger:EINterval:EVENT<x>:TYPE?  
<x> = 1 or 2

**Example** :TRIGGER:EINTERVAL:EVENT1:TYPE CANBUS  
:TRIGGER:EINTERVAL:EVENT1:TYPE?  
-> :TRIGGER:EINTERVAL:EVENT1:  
TYPE CANBUS

### **:TRIGger:EINterval:EVENT<x>:WIDTH?**

**Function** Queries all settings related to the pulse width trigger of the event.

**Syntax** :TRIGger:EINterval:EVENT<x>:WIDTH?  
<x> = 1 or 2

**Example** :TRIGGER:EINTERVAL:EVENT1:WIDTH?  
-> :TRIGGER:EINTERVAL:EVENT1:WIDTH:  
MODE TIMEOUT;POLARITY POSITIVE;  
SOURCE EXTERNAL;TIME1 1.000E+00;  
TIME2 1.000E+00



**:TRIGger:EINterval:EVENT<x>:WIDTH:MODE**

**Function** Sets the determination mode of the pulse width trigger or queries the current setting.

**Syntax** :TRIGger:EINterval:EVENT<x>:WIDTH:MODE {BETween|IN|NOTBetween|OUT|TIMEout}  
:TRIGger:EINterval:EVENT<x>:WIDTH:MODE?<x> = 1 or 2

**Example** :TRIGGER:EINTERVAL:EVENT1:WIDTH:MODE  
TIMEOUT  
:TRIGGER:EINTERVAL:EVENT1:WIDTH:MODE?  
-> :TRIGGER:EINTERVAL:EVENT1:WIDTH:MODE  
TIMEOUT

**:TRIGger:EINterval:EVENT<x>:WIDTH:POLarity**

**Function** Sets the polarity of the pulse width trigger or queries the current setting.

**Syntax** :TRIGger:EINterval:EVENT<x>:WIDTH:POLarity {FALSe|IN|NEGative|OUT|POSitive|TRUE}  
:TRIGger:EINterval:EVENT<x>:WIDTH:POLarity?<x> = 1 or 2

**Example** :TRIGGER:EINTERVAL:EVENT1:WIDTH:POLARITY POSITIVE  
:TRIGGER:EINTERVAL:EVENT1:WIDTH:POLARITY? -> :TRIGGER:EINTERVAL:EVENT1:WIDTH:POLARITY POSITIVE

**Description** • {IN|OUT} is valid when :TRIGger:EINterval:EVENT<x>:TYPE PQUALify|PULSE and :TRIGger:SOURce:CHANnel<x>:WINDow ON. {POSitive|NEGative} is valid when :TRIGger:EINterval:EVENT<x>:TYPE PQUALify|PULSE and :TRIGger:SOURce:CHANnel<x>:WINDow OFF.

- {FALSe|TRUE} is valid when :TRIGger:EINterval:EVENT<x>:TYPE PSTATE.

**:TRIGger:EINterval:EVENT<x>:WIDTH:SOURce**

**Function** Sets the trigger source of the pulse width trigger or queries the current setting.

**Syntax** :TRIGger:EINterval:EVENT<x>:WIDTH:SOURce {<NRf>|EXternal}  
:TRIGger:EINterval:EVENT<x>:WIDTH:SOURce?<x> = 1 or 2  
<NRf> = 1 to 4

**Example** :TRIGGER:EINTERVAL:EVENT1:WIDTH:SOURCE EXTERNAL  
:TRIGGER:EINTERVAL:EVENT1:WIDTH:SOURCE?  
-> :TRIGGER:EINTERVAL:EVENT1:WIDTH:SOURCE EXTERNAL

**Description** This command is valid when :TRIGger:EINterval:EVENT<x>:TYPE PQUALify|PULSE.

**:TRIGger:EINterval:EVENT<x>:WIDTH:TIME<x>**

**Function** Sets the pulse width of the pulse width trigger or queries the current setting.

**Syntax** :TRIGger:EINterval:EVENT<x>:WIDTH:TIME<x> {<Time>}  
:TRIGger:EINterval:EVENT<x>:WIDTH:TIME<x>?<x> of EVENT<x> = 1 or 2  
<x> of TIME<x> = 1 or 2  
<Time> = 1 ns to 10 s in 500-ps steps

**Example** :TRIGGER:EINTERVAL:EVENT1:WIDTH:TIME1 1S  
:TRIGGER:EINTERVAL:EVENT1:WIDTH:TIME1?  
-> :TRIGGER:EINTERVAL:EVENT1:WIDTH:TIME1 1.000E+00

**Description** TIME2 is valid when :TRIGger:EINterval:EVENT<x>:WIDTH:MODE BETWEEN|NOTBetween.

**:TRIGger:EINterval:MODE**

**Function** Sets the determination mode of the event interval or queries the current setting.

**Syntax** :TRIGger:EINterval:MODE {BETween|IN|NOTBetween|OUT|TIMEout}  
:TRIGger:EINterval:MODE?

**Example** :TRIGGER:EINTERVAL:MODE BETWEEN  
:TRIGGER:EINTERVAL:MODE? -> :TRIGGER:EINTERVAL:MODE BETWEEN

## 5.29 TRIGger Group

### **:TRIGger:EINterval:TIME<x>**

**Function** Sets the interval time of the event interval or queries the current setting.

**Syntax** :TRIGger:EINterval:TIME<x> {<Time>}  
:TRIGger:EINterval:TIME<x>?  
<x> = 1 or 2

<Time> = 1.5 ns to 10 s in 500-ps steps

**Example** :TRIGGER:EINTERVAL:TIME1 1S  
:TRIGGER:EINTERVAL:TIME1? -> :TRIGGER:  
EINTERVAL:TIME1 1.000E+00

**Description** This command is valid when :TRIGger:EINterval:  
MODE BETWEEN|NOTBetween.

### **:TRIGger:EINterval:TRY?**

**Function** Queries all settings related to the event interval trial.

**Syntax** :TRIGger:EINterval:TRY?

**Example** :TRIGGER:EINTERVAL:TRY? -> :TRIGGER:  
EINTERVAL:TRY:MODE 0;SELECT 1

### **:TRIGger:EINterval:TRY:MODE**

**Function** Sets the trial mode or queries the current setting.

**Syntax** :TRIGger:EINterval:TRY:MODE {<Boolean>}  
:TRIGger:EINterval:TRY:MODE?

**Example** :TRIGGER:EINTERVAL:TRY:MODE ON  
:TRIGGER:EINTERVAL:TRY:MODE?  
-> :TRIGGER:EINTERVAL:TRY:MODE 1

**Description** This command is valid when :TRIGger:  
EINterval:MODE BETWEEN|NOTBetween.

### **:TRIGger:EINterval:TRY:SElect**

**Function** Sets the source event of the trial mode or queries the current setting.

**Syntax** :TRIGger:EINterval:TRY:SElect {<NRF>}  
:TRIGger:EINterval:TRY:SElect?  
<NRF> = 1 or 2

**Example** :TRIGGER:EINTERVAL:TRY:SELECT 1  
:TRIGGER:EINTERVAL:TRY:SELECT?  
-> :TRIGGER:EINTERVAL:TRY:SELECT 1

**Description** This command is valid when :TRIGger:  
EINterval:MODE BETWEEN|NOTBetween.

### **:TRIGger:ENHanced?**

**Function** Queries all settings related to the enhanced trigger.

**Syntax** :TRIGger:ENHanced?

**Example** :TRIGGER:ENHANCED? -> :TRIGGER:  
ENHANCED:I2CBUS:ADATA:BIT10ADDRESS:  
PATTERN "00011111101";:TRIGGER:  
ENHANCED:I2CBUS:ADATA:BIT7ADDRESS:  
PATTERN "11011110";:TRIGGER:ENHANCED:  
I2CBUS:ADATA:BIT7APSUB:ADDRESS:  
PATTERN "11001101";:TRIGGER:ENHANCED:  
I2CBUS:ADATA:BIT7APSUB:SADDRESS:  
PATTERN "11101111";:TRIGGER:ENHANCED:  
I2CBUS:ADATA:TYPE BIT10ADDRESS;:  
TRIGGER:ENHANCED:I2CBUS:CLOCK:  
SOURCE 1;:TRIGGER:ENHANCED:I2CBUS:DATA:  
BYTE 1;CONDITION TRUE;DPOSITION 1;  
MODE 1;PATTERN1 "10101011";  
PATTERN2 "10001011";  
PATTERN3 "00101011";  
PATTERN4 "10101011";PMODE DONTCARE;  
SOURCE 1;:TRIGGER:ENHANCED:I2CBUS:  
GCALL:BIT7MADDRESS:PATTERN "1010101";:  
TRIGGER:ENHANCED:I2CBUS:GCALL:  
SBYTE BIT7MADDRESS;:TRIGGER:ENHANCED:  
I2CBUS:MODE ADATA;NAIGNORE:HSMODE 1;  
RACCESS 1;SBYTE 1;:TRIGGER:ENHANCED:  
I2CBUS:SBHSMODE:TYPE HSMODE;:TRIGGER:  
ENHANCED:SPATTERN:BITRATE 1.000E+00;  
CLOCK:MODE 1;POLARITY FALL;SOURCE 1;:  
TRIGGER:ENHANCED:SPATTERN:CS 1;DATA:  
ACTIVE HIGH;SOURCE 1;:TRIGGER:ENHANCED:  
SPATTERN:LATCH:SOURCE 1;POLARITY FALL;:  
TRIGGER:ENHANCED:SPATTERN:  
PATTERN "1100110111101111";:TRIGGER:  
ENHANCED:SPIBUS:BITORDER MSBFIRST;  
CLOCK:POLARITY FALL;SOURCE 1;:TRIGGER:  
ENHANCED:SPIBUS:CS:ACTIVE HIGH;  
SOURCE 1;:TRIGGER:ENHANCED:SPIBUS:  
DATA1:BYTE 1;CONDITION TRUE;  
DPOSITION 0;PATTERN1 "00010010";  
PATTERN2 "00110100";  
PATTERN3 "01010110";  
PATTERN4 "00010010";SOURCE 3;:  
TRIGGER:ENHANCED:SPIBUS:DATA2:BYTE 1;  
CONDITION TRUE;DPOSITION 0;  
PATTERN1 "00010010";  
PATTERN2 "00110100";  
PATTERN3 "01010110";  
PATTERN4 "00010010";SOURCE DONTCARE;:  
TRIGGER:ENHANCED:SPIBUS:MODE WIRE3.....

**:TRIGger:ENHanced:CANBus?**

**Function** Queries all settings related to the CAN bus signal trigger.

**Syntax** :TRIGger:ENHanced:CANBus?

**Example** :TRIGGER:ENHANCED:CANBUS? -> :TRIGGER:ENHANCED:CANBUS:  
ACK DONTCARE;BRATE 1000000;DATA:  
BORDER BIG;CONDITION DONTCARE;  
DATA1 0.0000000E+00;  
DATA2 255.000000E+00;DLC 8;MSBLSB 7,0;  
PATTERN "11100101011001000111100010  
0100110010101000100001000111111111  
010";SIGN UNSIGN;:TRIGGER:ENHANCED:  
CANBUS:IDEXT:PATTERN "XXXXXXXXXXXXXXXX  
XXXXXXXXXXXXXXXX";:TRIGGER:ENHANCED:  
CANBUS:IDOR:ID1:ACK DONTCARE;DATA:  
BORDER BIG;CONDITION DONTCARE;  
DATA1 0.0000000E+00;  
DATA2 255.000000E+00;DLC 8;MSBLSB 7,0;  
PATTERN "00000001001000110100010101  
1001111000100110101011110011011101  
111";SIGN UNSIGN;:TRIGGER:ENHANCED:  
CANBUS:IDOR:ID1:FORMAT STD;IDEXT:  
PATTERN "1101010111001101110111000  
0";:TRIGGER:ENHANCED:CANBUS:IDOR:ID1:  
IDSTD:PATTERN "00100100011";:TRIGGER:  
ENHANCED:CANBUS:IDOR:ID1:  
MODE 0;RTR DATA;:TRIGGER:ENHANCED:  
CANBUS:IDOR:ID2:ACK DONTCARE;DATA:  
BORDER BIG;CONDITION DONTCARE;  
DATA1 0.0000000E+00;  
DATA2 255.000000E+00;DLC 8;MSBLSB 7,0;  
PATTERN "1111110110111001011101010  
01100001110110010101000011001000010  
000";SIGN UNSIGN;:TRIGGER:ENHANCED:  
CANBUS:IDOR:ID2:FORMAT STD;IDEXT:  
PATTERN "1001000110100010101100111000  
";:TRIGGER:ENHANCED:CANBUS:IDOR:ID2:  
IDSTD:PATTERN "10001010110";:TRIGGER:  
ENHANCED:CANBUS:IDOR:ID2:MODE 0.....

**:TRIGger:ENHanced:CANBus:ACK**

**Function** Sets the ACK condition of the CAN bus signal trigger or queries the current setting.

**Syntax** :TRIGger:ENHanced:CANBus:ACK {ACK|ACKBoth|DONTcare|NONack}

:TRIGger:ENHanced:CANBus:ACK?

**Example** :TRIGGER:ENHANCED:CANBUS:ACK ACK  
:TRIGGER:ENHANCED:CANBUS:ACK?  
-> :TRIGGER:ENHANCED:CANBUS:ACK ACK

**:TRIGger:ENHanced:CANBus:BRATe**

**Function** Sets the bit rate (data transfer rate) of the CAN bus signal trigger or queries the current setting.

**Syntax** :TRIGger:ENHanced:CANBus:BRATe  
{<Nrf>|USER,<Nrf>}  
:TRIGger:ENHanced:CANBus:BRATe?  
<Nrf> = 33300, 83300, 125000, 250000, 500000,  
1000000  
<Nrf> of USER = See the User's Manual (IM701310-01E).

**Example** :TRIGGER:ENHANCED:CANBUS:BRATE 83300  
:TRIGGER:ENHANCED:CANBUS:BRATE?  
-> :TRIGGER:ENHANCED:CANBUS:BRATE 83300

**:TRIGger:ENHanced:CANBus:DATA?**

**Function** Queries all settings related to the CAN bus signal trigger data.

**Syntax** :TRIGger:ENHanced:CANBus:DATA?

**Example** :TRIGGER:ENHANCED:CANBUS:DATA?  
-> :TRIGGER:ENHANCED:CANBUS:DATA:  
BORDER BIG;CONDITION DONTCARE;  
DATA1 0.0000000E+00;  
DATA2 255.000000E+00;DLC 8;MSBLSB 7,0;  
PATTERN "1110010101100100011110001001  
00110010101000100001000111111111010"  
;SIGN UNSIGN

**:TRIGger:ENHanced:CANBus:DATA:BORDER**

**Function** Sets the byte order of the CAN bus signal trigger data or queries the current setting.

**Syntax** :TRIGger:ENHanced:CANBus:DATA:BORDER  
{BIG|LITtle}

:TRIGger:ENHanced:CANBus:DATA:BORDER?

**Example** :TRIGGER:ENHANCED:CANBUS:DATA:  
BORDER BIG  
:TRIGGER:ENHANCED:CANBUS:DATA:  
BORDER? -> :TRIGGER:ENHANCED:CANBUS:  
DATA:BORDER BIG

**:TRIGger:ENHanced:CANBus:DATA:CONDition**

**Function** Sets the data condition of the CAN bus signal trigger or queries the current setting.

**Syntax** :TRIGger:ENHanced:CANBus:DATA:  
CONDition {BETween|DONTcare|FALSe|  
GTHan|LTHan|ORANge|TRUE}  
:TRIGger:ENHanced:CANBus:DATA:  
CONDition?

**Example** :TRIGGER:ENHANCED:CANBUS:DATA:  
CONDITION BETWEEN  
:TRIGGER:ENHANCED:CANBUS:DATA:  
CONDITION? -> :TRIGGER:ENHANCED:  
CANBUS:DATA:CONDITION BETWEEN

## 5.29 TRIGger Group

### **:TRIGger:ENHanced:CANBus:DATA:DATA<x>**

**Function** Sets the comparison data of the CAN bus signal trigger data or queries the current setting.

**Syntax** :TRIGger:ENHanced:CANBus:DATA:DATA<x> {<NRf>}  
:TRIGger:ENHanced:CANBus:DATA:DATA<x>?<x> = 1 or 2  
<NRf> = See the User's Manual (IM701310-01E).

**Example** :TRIGGER:ENHANCED:CANBUS:DATA:DATA1 1  
:TRIGGER:ENHANCED:CANBUS:DATA:DATA1? -> :TRIGGER:ENHANCED:CANBUS:DATA:DATA1 1.0000000E+00

**Description** • Use :TRIGger:ENHANCED:CANBus:DATA:DATA1 when :TRIGger:ENHANCED:CANBus:DATA:CONDition GTHan is specified.  
• Use :TRIGger:ENHANCED:CANBus:DATA:DATA2 when :TRIGger:ENHANCED:CANBus:DATA:CONDition LTHan is specified.  
• Use :TRIGger:ENHANCED:CANBus:DATA:DATA1 to set the smaller value and :TRIGger:ENHANCED:CANBus:DATA:DATA2 to set the larger value when :TRIGger:ENHANCED:CANBus:DATA:CONDition BETween|ORANge is specified.

### **:TRIGger:ENHanced:CANBus:DATA:DLC**

**Function** Sets the number of valid bytes (DLC) of the CAN bus signal trigger data or queries the current setting.

**Syntax** :TRIGger:ENHanced:CANBus:DATA:DLC {<NRf>}  
:TRIGger:ENHanced:CANBus:DATA:DLC?<NRf> = 0 to 8

**Example** :TRIGGER:ENHANCED:CANBUS:DATA:DLC 0  
:TRIGGER:ENHANCED:CANBUS:DATA:DLC?  
-> :TRIGGER:ENHANCED:CANBUS:DATA:DLC 0

### **:TRIGger:ENHanced:CANBus:DATA:HEXA**

**Function** Sets the CAN bus signal trigger data in hexadecimal notation.

**Syntax** :TRIGger:ENHanced:CANBus:DATA:HEXA {<String>}  
<String> = Up to 16 characters by combining '0' to 'F' and 'X' (in one-byte unit)

**Example** :TRIGGER:ENHANCED:CANBUS:DATA:HEXA "A9"

### **:TRIGger:ENHanced:CANBus:DATA:MSBLsb**

**Function** Sets the MSB and LSB bits of the CAN bus signal trigger data or queries the current setting.

**Syntax** :TRIGger:ENHanced:CANBus:DATA:MSBLsb {<NRf>,<NRf>}  
:TRIGger:ENHanced:CANBus:DATA:MSBLsb?<NRf> = See the User's Manual (IM701310-01E).

**Example** :TRIGGER:ENHANCED:CANBUS:DATA:MSBLSB 1,0  
:TRIGGER:ENHANCED:CANBUS:DATA:MSBLSB? -> :TRIGGER:ENHANCED:CANBUS:DATA:MSBLSB 1,0

### **:TRIGger:ENHanced:CANBus:DATA:PATtern**

**Function** Sets the CAN bus signal trigger data in binary notation or queries the current setting.

**Syntax** :TRIGger:ENHanced:CANBus:DATA:PATtern {<String>}  
:TRIGger:ENHanced:CANBus:DATA:PATtern?<String> = Up to 64 characters by combining '0','1,' and 'X' (in one-byte unit)

**Example** :TRIGGER:ENHANCED:CANBUS:DATA:PATTERN "11011111"  
:TRIGGER:ENHANCED:CANBUS:DATA:PATTERN? -> :TRIGGER:ENHANCED:CANBUS:DATA:PATTERN "11011111"

### **:TRIGger:ENHanced:CANBus:DATA:SIGN**

**Function** Sets the sign of the CAN bus signal trigger data or queries the current setting.

**Syntax** :TRIGger:ENHanced:CANBus:DATA:SIGN {SIGN|UNSign}  
:TRIGger:ENHanced:CANBus:DATA:SIGN? Example :TRIGGER:ENHANCED:CANBUS:DATA:SIGN SIGN  
:TRIGGER:ENHANCED:CANBUS:DATA:SIGN? -> :TRIGGER:ENHANCED:CANBUS:DATA:SIGN SIGN

### **:TRIGger:ENHanced:CANBus:IDEXt?**

**Function** Queries all settings related to the ID of the extended format of the CAN bus signal trigger.

**Syntax** :TRIGger:ENHanced:CANBus:IDEXt?  
**Example** :TRIGGER:ENHANCED:CANBUS:IDEXT? -> :TRIGGER:ENHANCED:CANBUS:IDEXT:PATTERN "110010110111000011101110111111"

**:TRIGger:ENHanced:CANBus:IDEXT:HEXA**

**Function** Sets the ID of the extended format of the CAN bus signal trigger in hexadecimal notation.

**Syntax** :TRIGger:ENHanced:CANBus:IDEXT:  
HEXA {<String>}  
<String> = 8 characters by combining '0' to 'F' and 'X'

**Example** :TRIGGER:ENHANCED:CANBUS:IDEXT:  
HEXA "1AEF5906"

**:TRIGger:ENHanced:CANBus:IDEXT:PATtern**

**Function** Sets the ID of the extended format of the CAN bus signal trigger in binary notation or queries the current setting.

**Syntax** :TRIGger:ENHanced:CANBus:IDEXT:  
PATtern {<String>}  
:TRIGger:ENHanced:CANBus:IDEXT:  
PATtern?  
<String> = 29 characters by combining '0', '1,' and 'X'

**Example** :TRIGGER:ENHANCED:CANBUS:IDEXT:  
PATTERN "11001011011100001110111011111"  
:TRIGGER:ENHANCED:CANBUS:IDEXT:  
PATTERN? -> :TRIGGER:ENHANCED:CANBUS:  
IDEXT:PATTERN "1100101101110000111011  
1011111"

**:TRIGger:ENHanced:CANBus:IDOR?**

**Function** Queries all settings related to the OR condition of the CAN bus signal trigger.

**Syntax** :TRIGger:ENHanced:CANBus:IDOR?

**Example** :TRIGGER:ENHANCED:CANBUS:IDOR?  
-> :TRIGGER:ENHANCED:CANBUS:IDOR:ID1:  
ACK DONTCARE;DATA:BORDER BIG;  
CONDITION DONTCARE;  
DATA1 0.0000000E+00;  
DATA2 255.00000E+00;DLC 8;MSBLSB 7,0;  
PATTERN "00000001001000110100010101  
10011110001001101010111100110111101  
111";SIGN UNSIGN;:TRIGGER:ENHANCED:  
CANBUS:IDOR:ID1:FORMAT STD;IDEXT:  
PATTERN "110101011110011011101111000  
0";:TRIGGER:ENHANCED:CANBUS:IDOR:ID1:  
IDSTD:PATTERN "00100100011";:TRIGGER:  
ENHANCED:CANBUS:IDOR:ID1:  
MODE 0;RTR DATA;:TRIGGER:ENHANCED:  
CANBUS:IDOR:ID2:ACK DONTCARE;DATA:  
BORDER BIG;CONDITION DONTCARE;  
DATA1 0.0000000E+00;  
DATA2 255.00000E+00;DLC 8;MSBLSB 7,0;  
PATTERN "11111110110111001011101010  
01100001110110010101000011001000010  
000";SIGN UNSIGN;:TRIGGER:ENHANCED:  
CANBUS:IDOR:ID2:FORMAT STD;IDEXT:  
PATTERN "1001000110100010101100111100  
0";:TRIGGER:ENHANCED:CANBUS:IDOR:ID2:  
IDSTD:PATTERN "10001010110";:TRIGGER:  
ENHANCED:CANBUS:IDOR:ID2:  
MODE 0;RTR DATA;:TRIGGER:ENHANCED:  
CANBUS:IDOR:ID3:ACK DONTCARE;DATA:  
BORDER BIG;CONDITION DONTCARE;  
DATA1 0.0000000E+00;  
DATA2 255.00000E+00;DLC 8.....

**:TRIGger:ENHanced:CANBus:IDOR:ID<x>?**

**Function** Queries all settings related to each ID of the OR condition of the CAN bus signal trigger.

**Syntax** :TRIGger:ENHanced:CANBus:IDOR:ID<x>?  
<x> = 1 to 4

**Example** :TRIGGER:ENHANCED:CANBUS:IDOR:ID1?  
-> :TRIGGER:ENHANCED:CANBUS:IDOR:ID1:  
ACK DONTCARE;DATA:BORDER BIG;  
CONDITION DONTCARE;  
DATA1 0.0000000E+00;  
DATA2 255.00000E+00;DLC 8;MSBLSB 7,0;  
PATTERN "00000001001000110100010101  
10011110001001101010111100110111101  
111";SIGN UNSIGN;:TRIGGER:ENHANCED:  
CANBUS:IDOR:ID1:FORMAT STD;IDEXT:  
PATTERN "110101011110011011110111000  
0";:TRIGGER:ENHANCED:CANBUS:IDOR:ID1:  
IDSTD:PATTERN "00100100011";:TRIGGER:  
ENHANCED:CANBUS:IDOR:ID1:MODE 0;  
RTR DATA

## 5.29 TRIGger Group

### :TRIGger:ENHanced:CANBus:IDOR:ID<x>:

#### ACK

**Function** Sets each ACK condition of the OR condition of the CAN bus signal trigger or queries the current setting.

**Syntax** :TRIGger:ENHanced:CANBus:IDOR:ID<x>:  
ACK {ACK|ACKBoth|DONTcare|NONack}  
:TRIGger:ENHanced:CANBus:IDOR:ID<x>:  
ACK?  
<x> = 1 to 4

**Example** :TRIGGER:ENHANCED:CANBUS:IDOR:ID1:  
ACK ACK  
:TRIGGER:ENHANCED:CANBUS:IDOR:ID1:  
ACK? -> :TRIGGER:ENHANCED:CANBUS:  
IDOR:ID1:ACK ACK

### :TRIGger:ENHanced:CANBus:IDOR:ID<x>:

#### DATA?

**Function** Queries all settings related to each data of the OR condition of the CAN bus signal trigger.

**Syntax** :TRIGger:ENHanced:CANBus:IDOR:ID<x>:  
DATA?  
<x> = 1 to 4

**Example** :TRIGGER:ENHANCED:CANBUS:IDOR:ID1:  
DATA? -> :TRIGGER:ENHANCED:CANBUS:  
IDOR:ID1:DATA:BORDER BIG;  
CONDITION DONTCARE;  
DATA1 0.0000000E+00;  
DATA2 255.00000E+00;DLC 8;MSBLSB 7,0;  
PATTERN "00000001001000110100010101100  
11110001001101010111100110111101111"  
;SIGN UNSIGN

### :TRIGger:ENHanced:CANBus:IDOR:ID<x>:

#### DATA:BORDER

**Function** Sets byte order of each data of the OR condition of the CAN bus signal trigger or queries the current setting.

**Syntax** :TRIGger:ENHanced:CANBus:IDOR:ID<x>:  
DATA:BORDER {BIG|LITTLE}  
:TRIGger:ENHanced:CANBus:IDOR:ID<x>:  
DATA:BORDER?  
<x> = 1 to 4

**Example** :TRIGGER:ENHANCED:CANBUS:IDOR:ID1:  
DATA:BORDER BIG  
:TRIGGER:ENHANCED:CANBUS:IDOR:ID1:  
DATA:BORDER? -> :TRIGGER:ENHANCED:  
CANBUS:IDOR:ID1:DATA:BORDER BIG

### :TRIGger:ENHanced:CANBus:IDOR:ID<x>:

#### DATA:CONDITION

**Function** Sets each data condition of the OR condition of the CAN bus signal trigger or queries the current setting.

**Syntax** :TRIGger:ENHanced:CANBus:IDOR:ID<x>:  
DATA:CONDition {BETWEEN|DONTcare|  
FALSE|GTHan|LTHan|ORANge|TRUE}  
:TRIGger:ENHanced:CANBus:IDOR:ID<x>:  
DATA:CONDition?  
<x> = 1 to 4

**Example** :TRIGGER:ENHANCED:CANBUS:IDOR:ID1:  
DATA:CONDITION BETWEEN  
:TRIGGER:ENHANCED:CANBUS:IDOR:ID1:  
DATA:CONDITION? -> :TRIGGER:ENHANCED:  
CANBUS:IDOR:ID1:DATA:  
CONDITION BETWEEN

### :TRIGger:ENHanced:CANBus:IDOR:ID<x>:

#### DATA:DATA<x>

**Function** Sets comparison data of each data of the OR condition of the CAN bus signal trigger or queries the current setting.

**Syntax** :TRIGger:ENHanced:CANBus:IDOR:ID<x>:  
DATA:DATA<x> {<Nrf>}  
:TRIGger:ENHanced:CANBus:IDOR:ID<x>:  
DATA:DATA<x>?

<x> of ID<x> = 1 to 4  
<x> of DATA<x> = 1 or 2  
<NRF> = See the User's Manual (IM701310-01E).

**Example** :TRIGGER:ENHANCED:CANBUS:IDOR:ID1:  
DATA:DATA1 1  
:TRIGGER:ENHANCED:CANBUS:IDOR:ID1:  
DATA:DATA1? -> :TRIGGER:ENHANCED:  
CANBUS:IDOR:ID1:DATA:  
DATA1 1.0000000E+00

**Description**

- **Use** :TRIGger:ENHANCED:CANBus:IDOR:ID<x>:DATA:DATA1 when :TRIGger:ENHANCED:CANBus:IDOR:ID<x>:DATA:CONDition GTHan is specified.
- **Use** :TRIGger:ENHANCED:CANBus:IDOR:ID<x>:DATA:DATA2 when :TRIGger:ENHANCED:CANBus:IDOR:ID<x>:DATA:CONDition LTHan is specified.
- **Use** :TRIGger:ENHANCED:CANBus:IDOR:ID<x>:DATA:DATA1 to set the smaller value and :TRIGger:ENHANCED:CANBus:IDOR:ID<x>:DATA:DATA2 to set the larger value when :TRIGger:ENHANCED:CANBus:IDOR:ID<x>:DATA:CONDition BETWEEN|ORANge is specified.

**: TRIGger: ENHanced: CANBus: IDOR: ID<x>:****DATA: DLC**

**Function** Sets the number of valid bytes (DLC) of each data of the OR condition of the CAN bus signal trigger or queries the current setting.

**Syntax** :TRIGger:ENHanced:CANBus:IDOR:ID<x>:  
DATA:DLC {<Nrf>}  
:TRIGger:ENHanced:CANBus:IDOR:ID<x>:  
DATA:DLC?  
<x> = 1 to 4  
<Nrf> = 0 to 8

**Example** :TRIGGER:ENHANCED:CANBUS:IDOR:ID1:  
DATA:DLC 0  
:TRIGGER:ENHANCED:CANBUS:IDOR:ID1:  
DATA:DLC? -> :TRIGGER:ENHANCED:  
CANBUS:IDOR:ID1:DATA:DLC 0

**: TRIGger: ENHanced: CANBus: IDOR: ID<x>:****DATA: HEXA**

**Function** Sets each data of the OR condition of the CAN bus signal trigger in hexadecimal notation.

**Syntax** :TRIGger:ENHanced:CANBus:IDOR:ID<x>:  
DATA:HEXA {<String>}  
<x> = 1 to 4  
<String> = Up to 16 characters by combining '0' to 'F' and 'X' (in one-byte unit)

**Example** :TRIGGER:ENHANCED:CANBUS:IDOR:ID1:  
DATA:HEXA "A9"

**: TRIGger: ENHanced: CANBus: IDOR: ID<x>:****DATA: MSBLSb**

**Function** Sets the MSB and LSB bits of each data of the OR condition of the CAN bus signal trigger or queries the current setting.

**Syntax** :TRIGger:ENHanced:CANBus:IDOR:ID<x>:  
DATA:MSBLSb {<Nrf>,<Nrf>}  
:TRIGger:ENHanced:CANBus:IDOR:ID<x>:  
DATA:MSBLSb?  
<x> = 1 to 4  
<Nrf> = See the User's Manual (IM701310-01E).

**Example** :TRIGGER:ENHANCED:CANBUS:IDOR:ID1:  
DATA:MSBLSB 1,0  
:TRIGGER:ENHANCED:CANBUS:IDOR:ID1:  
DATA:MSBLSB? -> :TRIGGER:ENHANCED:  
CANBUS:IDOR:ID1:DATA:MSBLSB 1,0

**: TRIGger: ENHanced: CANBus: IDOR: ID<x>:****DATA: PATtern**

**Function** Sets each data of the OR condition of the CAN bus signal trigger in binary notation or queries the current setting.

**Syntax** :TRIGger:ENHanced:CANBus:IDOR:ID<x>:  
DATA:PATtern {<String>}  
:TRIGger:ENHanced:CANBus:IDOR:ID<x>:  
DATA:PATtern?  
<x> = 1 to 4  
<String> = Up to 64 characters by combining '0','1,' and 'X' (in one-byte unit)

**Example** :TRIGGER:ENHANCED:CANBUS:IDOR:ID1:  
DATA:PATTERN "11011111"  
:TRIGGER:ENHANCED:CANBUS:IDOR:ID1:  
DATA:PATTERN? -> :TRIGGER:ENHANCED:  
CANBUS:IDOR:ID1:DATA:PATTERN  
"11011111"

**: TRIGger: ENHanced: CANBus: IDOR: ID<x>:****DATA: SIGN**

**Function** Sets sign of each data of the OR condition of the CAN bus signal trigger or queries the current setting.

**Syntax** :TRIGger:ENHanced:CANBus:IDOR:ID<x>:  
DATA:SIGN {SIGN|UNSign}  
:TRIGger:ENHanced:CANBus:IDOR:ID<x>:  
DATA:SIGN?  
<x> = 1 to 4

**Example** :TRIGGER:ENHANCED:CANBUS:IDOR:ID1:  
DATA:SIGN SIGN  
:TRIGGER:ENHANCED:CANBUS:IDOR:ID1:  
DATA:SIGN? -> :TRIGGER:ENHANCED:  
CANBUS:IDOR:ID1:DATA:SIGN SIGN

**: TRIGger: ENHanced: CANBus: IDOR: ID<x>:****FORMat**

**Function** Sets each message format (standard or extended) of the OR condition of the CAN bus signal trigger or queries the current setting.

**Syntax** :TRIGger:ENHanced:CANBus:IDOR:ID<x>:  
FORMat {STD|EXT}  
:TRIGger:ENHanced:CANBus:IDOR:ID<x>:  
FORMat?  
<x> = 1 to 4

**Example** :TRIGGER:ENHANCED:CANBUS:IDOR:ID1:  
FORMAT STD  
:TRIGGER:ENHANCED:CANBUS:IDOR:ID1:  
FORMAT? -> :TRIGGER:ENHANCED:CANBUS:  
IDOR:ID1:FORMAT STD

## 5.29 TRIGger Group

### **:TRIGger:ENHanced:CANBus:IDOR:ID<x>:IDEXt?**

**Function** Queries all settings related to the ID of each extended format of the OR condition of the CAN bus signal trigger.

**Syntax** :TRIGger:ENHanced:CANBus:IDOR:ID<x>:  
IDEXt?  
<x> = 1 to 4

**Example** :TRIGGER:ENHANCED:CANBUS:IDOR:ID1:  
IDEXT? -> :TRIGGER:ENHANCED:CANBUS:  
IDOR:ID1:IDEXT:PATTERN "1100101101110  
0001110111011111"

### **:TRIGger:ENHanced:CANBus:IDOR:ID<x>:IDEXt:HEXA**

**Function** Sets the ID of each extended format of the OR condition of the CAN bus signal trigger in hexadecimal notation.

**Syntax** :TRIGger:ENHanced:CANBus:IDOR:ID<x>:  
IDEXt:HEXA {<String>}  
<x> = 1 to 4

<String> = 8 characters by combining '0' to 'F' and 'X'  
**Example** :TRIGGER:ENHANCED:CANBUS:IDOR:ID1:  
IDEXT:HEXA "1AEF5906"

### **:TRIGger:ENHanced:CANBus:IDOR:ID<x>:IDEXt:PATtern**

**Function** Sets the ID of each extended format of the OR condition of the CAN bus signal trigger in binary notation or queries the current setting.

**Syntax** :TRIGger:ENHanced:CANBus:IDOR:ID<x>:  
IDEXt:PATtern {<String>}  
:TRIGger:ENHanced:CANBus:IDOR:ID<x>:  
IDEXt:PATtern?  
<x> = 1 to 4

<String> = 29 characters by combining '0','1,' and 'X'  
**Example** :TRIGGER:ENHANCED:CANBUS:IDOR:ID1:  
IDEXT:PATTERN "1100101101110000111011  
10111111"  
:TRIGGER:ENHANCED:CANBUS:IDOR:ID1:  
IDEXT:PATTERN? -> :TRIGGER:ENHANCED:  
CANBUS:IDOR:ID1:IDEXT:PATTERN "110010  
11011100001110111011111"

### **:TRIGger:ENHanced:CANBus:IDOR:ID<x>:IDSTd?**

**Function** Queries all settings related to the ID of each standard format of the OR condition of the CAN bus signal trigger.

**Syntax** :TRIGger:ENHanced:CANBus:IDOR:ID<x>:  
IDSTd?  
<x> = 1 to 4

**Example** :TRIGGER:ENHANCED:CANBUS:IDOR:ID1:  
IDSTD? -> :TRIGGER:ENHANCED:CANBUS:  
IDOR:ID1:IDSTD:PATTERN "000111111101"

### **:TRIGger:ENHanced:CANBus:IDOR:ID<x>:IDSTd:HEXA**

**Function** Sets the ID of each standard format of the OR condition of the CAN bus signal trigger in hexadecimal notation.

**Syntax** :TRIGger:ENHanced:CANBus:IDOR:ID<x>:  
IDSTd:HEXA {<String>}  
<x> = 1 to 4

<String> = 3 characters by combining '0' to 'F' and 'X'  
**Example** :TRIGGER:ENHANCED:CANBUS:IDOR:ID1:  
IDSTD:HEXA "5DF"

### **:TRIGger:ENHanced:CANBus:IDOR:ID<x>:IDSTd:PATtern**

**Function** Sets the ID of each standard format of the OR condition of the CAN bus signal trigger in binary notation or queries the current setting.

**Syntax** :TRIGger:ENHanced:CANBus:IDOR:ID<x>:  
IDSTd:PATtern {<String>}  
:TRIGger:ENHanced:CANBus:IDOR:ID<x>:  
IDSTd:PATtern?  
<x> = 1 to 4

<String> = 11 characters by combining '0','1,' and 'X'  
**Example** :TRIGGER:ENHANCED:CANBUS:IDOR:ID1:  
IDSTD:PATTERN "10111011111"  
:TRIGGER:ENHANCED:CANBUS:IDOR:ID1:  
IDSTD:PATTERN? -> :TRIGGER:ENHANCED:  
CANBUS:IDOR:ID1:IDSTD:PATTERN  
"10111011111"



**:TRIGger:ENHanced:CANBus:IDOR:ID<x>:****MODE**

**Function** Enables or disables each condition of the OR condition of the CAN bus signal trigger or queries the current setting.

**Syntax** :TRIGger:ENHanced:CANBus:IDOR:ID<x>:  
MODE {<Boolean>}  
:TRIGger:ENHanced:CANBus:IDOR:ID<x>:  
MODE?  
<x> = 1 to 4

**Example** :TRIGGER:ENHANCED:CANBUS:IDOR:ID1:  
MODE ON  
:TRIGGER:ENHANCED:CANBUS:IDOR:ID1:  
MODE? -> :TRIGGER:ENHANCED:CANBUS:  
IDOR:ID1:MODE 1

**:TRIGger:ENHanced:CANBus:IDOR:ID<x>:****RTR**

**Function** Sets each RTR of the OR condition of the CAN bus signal trigger or queries the current setting.

**Syntax** :TRIGger:ENHanced:CANBus:IDOR:ID<x>:  
RTR {DATA|DONTcare|REMOte}  
:TRIGger:ENHanced:CANBus:IDOR:ID<x>:  
RTR?  
<x> = 1 to 4

**Example** :TRIGGER:ENHANCED:CANBUS:IDOR:ID1:  
RTR DATA  
:TRIGGER:ENHANCED:CANBUS:IDOR:ID1:  
RTR? -> :TRIGGER:ENHANCED:CANBUS:  
IDOR:ID1:RTR DATA

**:TRIGger:ENHanced:CANBus:IDSTd?**

**Function** Queries all settings related to the ID of the standard format of the CAN bus signal trigger.

**Syntax** :TRIGger:ENHanced:CANBus:IDSTd?

**Example** :TRIGGER:ENHANCED:CANBUS:IDSTD?  
-> :TRIGGER:ENHANCED:CANBUS:IDSTD:  
PATTERN "00011111101"

**:TRIGger:ENHanced:CANBus:IDSTd:HEXA**

**Function** Sets the ID of the standard format of the CAN bus signal trigger in hexadecimal notation.

**Syntax** :TRIGger:ENHanced:CANBus:IDSTd:HEXA  
{<String>}

<String> = 3 characters by combining '0' to 'F' and 'X'  
**Example** :TRIGGER:ENHANCED:CANBUS:IDSTD:HEXA  
"5DF"

**:TRIGger:ENHanced:CANBus:IDSTd:PATtern**

**Function** Sets the ID of the standard format of the CAN bus signal trigger in binary notation or queries the current setting.

**Syntax** :TRIGger:ENHanced:CANBus:IDSTd:  
PATtern {<String>}  
:TRIGger:ENHanced:CANBus:IDSTd:  
PATtern?  
<String> = 11 characters by combining '0,' '1,' and 'X'

**Example** :TRIGGER:ENHANCED:CANBUS:IDSTD:  
PATTERN "10111011111"  
:TRIGGER:ENHANCED:CANBUS:IDSTD:  
PATTERN? -> :TRIGGER:ENHANCED:CANBUS:  
IDSTD:PATTERN "10111011111"

**:TRIGger:ENHanced:CANBus:MODE**

**Function** Sets the CAN bus signal trigger mode or queries the current setting.

**Syntax** :TRIGger:ENHanced:CANBus:MODE {EFrame|  
IDExt|IDOR|IDSTd|SOE}  
:TRIGger:ENHanced:CANBus:MODE?

**Example** :TRIGGER:ENHANCED:CANBUS:MODE EFRAME  
:TRIGGER:ENHANCED:CANBUS:MODE?  
-> :TRIGGER:ENHANCED:CANBUS:MODE EFRAME

**:TRIGger:ENHanced:CANBus:REcessive**

**Function** Sets the recessive level (bus level) of the CAN bus signal trigger or queries the current setting.

**Syntax** :TRIGger:ENHanced:CANBus:  
REcessive {HIGH|LOW}  
:TRIGger:ENHanced:CANBus:REcessive?

**Example** :TRIGGER:ENHANCED:CANBUS:  
RECESSIVE HIGH  
:TRIGGER:ENHANCED:CANBUS:  
RECESSIVE? -> :TRIGGER:ENHANCED:  
CANBUS:RECESSIVE HIGH

**:TRIGger:ENHanced:CANBus:RTR**

**Function** Sets the RTR of the CAN bus signal trigger or queries the current setting.

**Syntax** :TRIGger:ENHanced:CANBus:  
RTR {DATA|DONTcare|REMOte}  
:TRIGger:ENHanced:CANBus:RTR?

**Example** :TRIGGER:ENHANCED:CANBUS:RTR DATA  
:TRIGGER:ENHANCED:CANBUS:RTR?  
-> :TRIGGER:ENHANCED:CANBUS:RTR DATA

## 5.29 TRIGger Group

### **:TRIGger:ENHanced:CANBus:SOURce**

**Function** Sets the trigger source of the CAN bus signal trigger or queries the current setting.

**Syntax** :TRIGger:ENHanced:CANBus:SOURce  
{<NRf>}  
:TRIGger:ENHanced:CANBus:SOURce?  
<NRf> = 1 to 4

**Example** :TRIGGER:ENHANCED:CANBUS:SOURCE 1  
:TRIGGER:ENHANCED:CANBUS:SOURCE?  
-> :TRIGGER:ENHANCED:CANBUS:SOURCE 1

### **:TRIGger:ENHanced:CANBus:SPOint**

**Function** Sets the sample point of the CAN bus signal trigger or queries the current setting.

**Syntax** :TRIGger:ENHanced:CANBus:SPOint  
{<NRf>}  
:TRIGger:ENHanced:CANBus:SPOint?  
<NRf> = 18.8 to 90.6(%)

**Example** :TRIGGER:ENHANCED:CANBUS:SPOINT 18.8  
:TRIGGER:ENHANCED:CANBUS:  
SPOINT? -> :TRIGGER:ENHANCED:CANBUS:  
SPOINT 18.8E+00

### **:TRIGger:ENHanced:I2CBus?**

**Function** Queries all settings related to the I<sup>2</sup>C bus trigger.

**Syntax** :TRIGger:ENHanced:I2CBus?

**Example** :TRIGGER:ENHANCED:I2CBUS? -> :TRIGGER:  
ENHANCED:I2CBUS:ADATA:  
BIT10ADDRESS:PATTERN "10111011111";:  
TRIGGER:ENHANCED:I2CBUS:ADATA:  
BIT7ADDRESS:PATTERN "11011110";:  
TRIGGER:ENHANCED:I2CBUS:ADATA:  
BIT7APSUB:ADDRESS:PATTERN "10101011";:  
TRIGGER:ENHANCED:I2CBUS:ADATA:  
BIT7APSUB:SADDRESS:PATTERN "10101011";:  
TRIGGER:ENHANCED:I2CBUS:ADATA:  
TYPE BIT10ADDRESS;:TRIGGER:ENHANCED:  
I2CBUS:CLOCK:SOURCE 1;:TRIGGER:  
ENHANCED:I2CBUS:DATA:BYTE 1;  
CONDITION TRUE;DPOSITION 1;MODE 1;  
PATTERN1 "10101011";  
PATTERN2 "10101010";  
PATTERN3 "10101111";  
PATTERN4 "10101011";PMODE DONTCARE;  
SOURCE 1;:TRIGGER:ENHANCED:I2CBUS:  
GCALL:BIT7MADDRESS:PATTERN "1010101";:  
TRIGGER:ENHANCED:I2CBUS:GCALL:  
SBYTE BIT7MADDRESS;:TRIGGER:ENHANCED:  
I2CBUS:MODE ADATA;NAIGNORE:HSMODE 1;  
RACCESS 1;SBYTE 1;:TRIGGER:ENHANCED:  
I2CBUS:SBHSMODE:TYPE HSMODE

### **:TRIGger:ENHanced:I2CBus:ADATa?**

**Function** Queries all settings related to the address of the I<sup>2</sup>C bus trigger.

**Syntax** :TRIGger:ENHanced:I2CBus:ADATa?  
**Example** :TRIGGER:ENHANCED:I2CBUS:ADATa?

```
-> :TRIGGER:ENHANCED:I2CBUS:ADATa:
BIT10ADDRESS:PATTERN "10111011111";:
TRIGGER:ENHANCED:I2CBUS:ADATa:
BIT7ADDRESS:PATTERN "11011110";:
TRIGGER:ENHANCED:I2CBUS:ADATa:
BIT7APSUB:ADDRESS:PATTERN "10101011";:
TRIGGER:ENHANCED:I2CBUS:ADATa:
BIT7APSUB:SADDRESS:PATTERN "10101011";:
TRIGGER:ENHANCED:I2CBUS:ADATa:
TYPE BIT10ADDRESS
```

### **:TRIGger:ENHanced:I2CBus:ADATa:BIT10address?**

**Function** Queries all settings related to the 10-bit address of the I<sup>2</sup>C bus trigger.

**Syntax** :TRIGger:ENHanced:I2CBus:ADATa:  
BIT10address?

**Example** :TRIGGER:ENHANCED:I2CBUS:ADATa:  
BIT10ADDRESS? -> :TRIGGER:ENHANCED:  
I2CBUS:ADATa:BIT10ADDRESS:  
PATTERN "10111011111"

### **:TRIGger:ENHanced:I2CBus:ADATa:BIT10address:HEXA**

**Function** Sets the 10-bit address of the I<sup>2</sup>C bus trigger in hexadecimal notation.

**Syntax** :TRIGger:ENHanced:I2CBus:ADATa:  
BIT10address:HEXA {<String>}

<String> = 3 characters by combining '0' to 'F' and 'X'  
(bit 8 is the R $\bar{W}$  bit)

**Example** :TRIGGER:ENHANCED:I2CBUS:ADATa:  
BIT10ADDRESS:HEXA "7AB"

### **:TRIGger:ENHanced:I2CBus:ADATa:BIT10address:PATtern**

**Function** Sets the 10-bit address of the I<sup>2</sup>C bus trigger in binary notation or queries the current setting.

**Syntax** :TRIGger:ENHanced:I2CBus:ADATa:  
BIT10address:PATtern {<String>}  
:TRIGger:ENHanced:I2CBus:ADATa:  
BIT10address:PATtern?

<String> = 11 characters by combining '0', '1', and 'X'  
(bit 8 is the R $\bar{W}$  bit)

**Example** :TRIGGER:ENHANCED:I2CBUS:ADATa:  
BIT10ADDRESS:PATTERN "10111011111"  
:TRIGGER:ENHANCED:I2CBUS:ADATa:  
BIT10ADDRESS:PATTERN? -> :TRIGGER:  
ENHANCED:I2CBUS:ADATa:  
BIT10ADDRESS:PATTERN "10111011111"

**:TRIGger:ENHanced:I2CBus:ADATa:BIT7Address?**

Function Queries all settings related to the 7-bit address of the I<sup>2</sup>C bus trigger.

Syntax :TRIGger:ENHanced:I2CBus:ADATa:BIT7Address?

Example :TRIGGER:ENHANCED:I2CBUS:ADATA:BIT7ADDRESS? ->  
:TRIGGER:ENHANCED:I2CBUS:ADATA:BIT7ADDRESS:PATTERN "11011110"

**:TRIGger:ENHanced:I2CBus:ADATa:BIT7Address:HEXA**

Function Sets the 7-bit address of the I<sup>2</sup>C bus trigger in hexadecimal notation.

Syntax :TRIGger:ENHanced:I2CBus:ADATa:BIT7Address:HEXA {<String>}  
<String> = 2 characters by combining '0' to 'F' and 'X' (bit 0 is the R/W bit)

Example :TRIGGER:ENHANCED:I2CBUS:ADATA:BIT7ADDRESS:HEXA "DE"

**:TRIGger:ENHanced:I2CBus:ADATa:BIT7Address:PATtern**

Function Sets the 7-bit address of the I<sup>2</sup>C bus trigger in binary notation or queries the current setting.

Syntax :TRIGger:ENHanced:I2CBus:ADATa:BIT7Address:PATtern {<String>}  
:TRIGger:ENHanced:I2CBus:ADATa:BIT7Address:PATtern?  
<String> = 8 characters by combining '0', '1', and 'X' (bit 0 is the R/W\_bit)

Example :TRIGGER:ENHANCED:I2CBUS:ADATA:BIT7ADDRESS:PATTERN "11011110"  
:TRIGGER:ENHANCED:I2CBUS:ADATA:BIT7ADDRESS:PATTERN? -> :TRIGGER:ENHANCED:I2CBUS:ADATA:BIT7ADDRESS:PATTERN "11011110"

**:TRIGger:ENHanced:I2CBus:ADATa:BIT7APsub?**

Function Queries all settings related to the 7-bit + Sub address of the I<sup>2</sup>C bus trigger.

Syntax :TRIGger:ENHanced:I2CBus:ADATa:BIT7APsub?

Example :TRIGGER:ENHANCED:I2CBUS:ADATA:BIT7APSUB? -> :TRIGGER:ENHANCED:I2CBUS:ADATA:BIT7APSUB:ADDRESS:PATTERN "10101011"; :TRIGGER:ENHANCED:I2CBUS:ADATA:BIT7APSUB:SADDRESS:PATTERN "10101011"

**:TRIGger:ENHanced:I2CBus:ADATa:BIT7APsub:ADDRESS?**

Function Queries all settings related to the 7-bit address of the 7-bit + Sub address of the I<sup>2</sup>C bus trigger.

Syntax :TRIGger:ENHanced:I2CBus:ADATa:BIT7APsub:ADDRESS?

Example :TRIGGER:ENHANCED:I2CBUS:ADATA:BIT7APSUB:ADDRESS? -> :TRIGGER:ENHANCED:I2CBUS:ADATA:BIT7APSUB:ADDRESS:PATTERN "10101011"

**:TRIGger:ENHanced:I2CBus:ADATa:BIT7APsub:ADDRESS:HEXA**

Function Sets the 7-bit address of the 7-bit + Sub address of the I<sup>2</sup>C bus trigger in hexadecimal notation.

Syntax :TRIGger:ENHanced:I2CBus:ADATa:BIT7APsub:ADDRESS:HEXA {<String>}  
<String> = 2 characters by combining '0' to 'F' and 'X' (bit 0 is the R/W bit)

Example :TRIGGER:ENHANCED:I2CBUS:ADATA:BIT7APSUB:ADDRESS:HEXA "AB"

**:TRIGger:ENHanced:I2CBus:ADATa:BIT7APsub:ADDRESS:PATtern**

Function Sets the 7-bit address of the 7-bit + Sub address of the I<sup>2</sup>C bus trigger in binary notation or queries the current setting.

Syntax :TRIGger:ENHanced:I2CBus:ADATa:BIT7APsub:ADDRESS:PATtern {<String>}  
:TRIGger:ENHanced:I2CBus:ADATa:BIT7APsub:ADDRESS:PATtern?  
<String> = 8 characters by combining '0', '1', and 'X' (bit 0 is the R/W bit)

Example :TRIGGER:ENHANCED:I2CBUS:ADATA:BIT7APSUB:ADDRESS:PATTERN "10101011"  
:TRIGGER:ENHANCED:I2CBUS:ADATA:BIT7APSUB:ADDRESS:PATTERN? -> :TRIGGER:ENHANCED:I2CBUS:ADATA:BIT7APSUB:ADDRESS:PATTERN "10101011"

**:TRIGger:ENHanced:I2CBus:ADATa:BIT7APsub:SADDRESS?**

Function Queries all settings related to the Sub address of the 7-bit + Sub address of the I<sup>2</sup>C bus trigger.

Syntax :TRIGger:ENHanced:I2CBus:ADATa:BIT7APsub:SADDRESS?

Example :TRIGGER:ENHANCED:I2CBUS:ADATA:BIT7APSUB:SADDRESS? -> :TRIGGER:ENHANCED:I2CBUS:ADATA:BIT7APSUB:SADDRESS:PATTERN "10101011"

## 5.29 TRIGger Group

### **:TRIGger:ENHanced:I2Cbus:ADATa:**

#### **BIT7APsub:SADdress:HEXA**

**Function** Sets the Sub address of the 7-bit + Sub address of the I<sup>2</sup>C bus trigger in hexadecimal notation.

**Syntax** :TRIGger:ENHanced:I2Cbus:ADATa:  
BIT7APsub:SADdress:HEXA {<String>}  
<String> = 2 characters by combining '0' to 'F' and 'X'

**Example** :TRIGGER:ENHANCED:I2CBUS:ADATA:  
BIT7APSUB:SADDRESS:HEXA "EF"

### **:TRIGger:ENHanced:I2Cbus:ADATa:**

#### **BIT7APsub:SADdress:PATtern**

**Function** Sets the Sub address of the 7-bit + Sub address of the I<sup>2</sup>C bus trigger in binary notation or queries the current setting.

**Syntax** :TRIGger:ENHanced:I2Cbus:ADATa:  
BIT7APsub:SADdress:PATtern {<String>}  
:TRIGger:ENHanced:I2Cbus:ADATa:  
BIT7APsub:SADdress:PATtern?  
<String> = 8 characters by combining '0','1,' and 'X'

**Example** :TRIGGER:ENHANCED:I2CBUS:ADATA:  
BIT7APSUB:SADDRESS:PATTERN "10101011"  
:TRIGGER:ENHANCED:I2CBUS:ADATA:  
BIT7APSUB:SADDRESS:PATTERN? -> :  
TRIGGER:ENHANCED:I2CBUS:ADATA:  
BIT7APSUB:SADDRESS:PATTERN "10101011"

### **:TRIGger:ENHanced:I2Cbus:ADATa:TYPE**

**Function** Sets the address type of the I<sup>2</sup>C bus trigger or queries the current setting.

**Syntax** :TRIGger:ENHanced:I2Cbus:ADATa:  
TYPE {BIT10address|BIT7Address|  
BIT7APsub}  
:TRIGger:ENHanced:I2Cbus:ADATa:TYPE?

**Example** :TRIGGER:ENHANCED:I2CBUS:ADATA:  
TYPE BIT10ADDRESS  
:TRIGGER:ENHANCED:I2CBUS:ADATA:TYPE?  
-> :TRIGGER:ENHANCED:I2CBUS:ADATA:  
TYPE BIT10ADDRESS

### **:TRIGger:ENHanced:I2Cbus:CLOCK?**

**Function** Queries all settings related to the clock of the I<sup>2</sup>C bus trigger.

**Syntax** :TRIGger:ENHanced:I2Cbus:CLOCK?

**Example** :TRIGGER:ENHANCED:I2CBUS:CLOCK?  
-> :TRIGGER:ENHANCED:I2CBUS:CLOCK:  
SOURCE 1

### **:TRIGger:ENHanced:I2Cbus:CLOCK:SOURce**

**Function** Sets the clock trace of the I<sup>2</sup>C bus trigger or queries the current setting.

**Syntax** :TRIGger:ENHanced:I2Cbus:CLOCK:  
SOURce {<Nrf>}  
:TRIGger:ENHanced:I2Cbus:CLOCK:SOURce?  
<Nrf> = 1 to 4

**Example** :TRIGGER:ENHANCED:I2CBUS:CLOCK:SOURCE 1  
:TRIGGER:ENHANCED:I2CBUS:CLOCK:SOURCE?  
-> :TRIGGER:ENHANCED:I2CBUS:CLOCK:  
SOURCE 1

### **:TRIGger:ENHanced:I2Cbus:DATA?**

**Function** Queries all settings related to the data of the I<sup>2</sup>C bus trigger.

**Syntax** :TRIGger:ENHanced:I2Cbus:DATA?  
<x> = 1 or 2

**Example** :TRIGGER:ENHANCED:I2CBUS:DATA?  
-> :TRIGGER:ENHANCED:I2CBUS:DATA:BYTE 1;  
CONDITION TRUE;DPOSITION 1;MODE 1;  
PATTERN1 "10101011";  
PATTERN2 "10101010";  
PATTERN3 "10101111";  
PATTERN4 "10101011";PMODE DONTCARE;  
SOURCE 1

### **:TRIGger:ENHanced:I2Cbus:DATA:BYTE**

**Function** Sets the number of data bytes of the I<sup>2</sup>C bus trigger or queries the current setting.

**Syntax** :TRIGger:ENHanced:I2Cbus:DATA:  
BYTE {<Nrf>}  
:TRIGger:ENHanced:I2Cbus:DATA:BYTE?  
<Nrf> = 1 to 4

**Example** :TRIGGER:ENHANCED:I2CBUS:DATA:BYTE 1  
:TRIGGER:ENHANCED:I2CBUS:DATA:BYTE?  
-> :TRIGGER:ENHANCED:I2CBUS:DATA:BYTE 1

### **:TRIGger:ENHanced:I2Cbus:DATA:**

#### **CONDition**

**Function** Sets the determination method (match or not match) of the data of the I<sup>2</sup>C bus trigger or queries the current setting.

**Syntax** :TRIGger:ENHanced:I2Cbus:DATA:  
CONDition {FALSe|TRUE}  
:TRIGger:ENHanced:I2Cbus:DATA:  
CONDition?

**Example** :TRIGGER:ENHANCED:I2CBUS:DATA:  
CONDITION TRUE  
:TRIGGER:ENHANCED:I2CBUS:DATA:  
CONDITION? -> :TRIGGER:ENHANCED:I2CBUS:  
DATA:CONDITION TRUE

**:TRIGger:ENHanced:I2Cbus:DATA:****DPOsition**

**Function** Sets the position for comparing the data pattern of the I<sup>2</sup>C bus trigger or queries the current setting.

**Syntax** :TRIGger:ENHanced:I2Cbus:DATA:  
DPOsition {<NRf>}  
:TRIGger:ENHanced:I2Cbus:DATA:  
DPOsition?  
<NRf> = 0 to 9999

**Example** :TRIGGER:ENHANCED:I2CBUS:DATA:  
DPOSITION 1  
:TRIGGER:ENHANCED:I2CBUS:DATA:  
DPOSITION? -> :TRIGGER:ENHANCED:I2CBUS:  
DATA:DPOSITION 1

**:TRIGger:ENHanced:I2Cbus:DATA:HEXA<x>**

**Function** Sets the data of the I<sup>2</sup>C bus trigger in hexadecimal notation.

**Syntax** :TRIGger:ENHanced:I2Cbus:DATA:  
HEXA<x> {<String>}  
<x> = 1 to 4

<String> = 2 characters by combining '0' to 'F' and 'X'

**Example** :TRIGGER:ENHANCED:I2CBUS:DATA:  
HEXA1 "AB"

**:TRIGger:ENHanced:I2Cbus:DATA:MODE**

**Function** Enables/Disables the data conditions of the I<sup>2</sup>C bus trigger or queries the current setting.

**Syntax** :TRIGger:ENHanced:I2Cbus:DATA:  
MODE {<Boolean>}  
:TRIGger:ENHanced:I2Cbus:DATA:MODE?

**Example** :TRIGGER:ENHANCED:I2CBUS:DATA:MODE ON  
:TRIGGER:ENHANCED:I2CBUS:DATA:MODE?  
-> :TRIGGER:ENHANCED:I2CBUS:DATA:MODE 1

**:TRIGger:ENHanced:I2Cbus:DATA:****PATtern<x>**

**Function** Sets the data of the I<sup>2</sup>C bus trigger in binary notation or queries the current setting.

**Syntax** :TRIGger:ENHanced:I2Cbus:DATA:  
PATtern<x> {<String>}  
:TRIGger:ENHanced:I2Cbus:DATA:  
PATtern<x>?  
<x> = 1 to 4

<String> = 8 characters by combining '0', '1,' and 'X'

**Example** :TRIGGER:ENHANCED:I2CBUS:DATA:  
PATTERN1 "10101011"  
:TRIGGER:ENHANCED:I2CBUS:DATA:PATTERN1?  
-> :TRIGGER:ENHANCED:I2CBUS:DATA:  
PATTERN1 "10101011"

**:TRIGger:ENHanced:I2Cbus:DATA:PMODE**

**Function** Sets the pattern comparison start position mode of the data of the I<sup>2</sup>C bus trigger or queries the current setting.

**Syntax** :TRIGger:ENHanced:I2Cbus:DATA:  
PMODE {DONTcare|SElect}  
:TRIGger:ENHanced:I2Cbus:DATA:PMODE?

**Example** :TRIGGER:ENHANCED:I2CBUS:DATA:  
PMODE SELECT  
:TRIGGER:ENHANCED:I2CBUS:DATA:PMODE?  
-> :TRIGGER:ENHANCED:I2CBUS:DATA:  
PMODE SELECT

**:TRIGger:ENHanced:I2Cbus:DATA:SOURce**

**Function** Sets the data trace of the I<sup>2</sup>C bus trigger or queries the current setting.

**Syntax** :TRIGger:ENHanced:I2Cbus:DATA:  
SOURce {<NRf>}  
:TRIGger:ENHanced:I2Cbus:DATA:SOURce?  
<NRf> = 1 to 4

**Example** :TRIGGER:ENHANCED:I2CBUS:DATA:SOURCE 1  
:TRIGGER:ENHANCED:I2CBUS:DATA:SOURCE?  
-> :TRIGGER:ENHANCED:I2CBUS:DATA:  
SOURCE 1

**:TRIGger:ENHanced:I2Cbus:GCALl?**

**Function** Queries all settings related to the general call of the I<sup>2</sup>C bus trigger.

**Syntax** :TRIGger:ENHanced:I2Cbus:GCALl?  
<x> = 1 or 2

**Example** :TRIGGER:ENHANCED:I2CBUS:GCALL?  
-> :TRIGGER:ENHANCED:I2CBUS:GCALL:  
BIT7MADDRESS:PATTERN "1010101";:  
TRIGGER:ENHANCED:I2CBUS:GCALL:  
SBYTE BIT7MADDRESS

**:TRIGger:ENHanced:I2Cbus:GCALl:**  
**BIT7maddress?**

**Function** Queries all settings related to the 7-bit master address of the general call of the I<sup>2</sup>C bus trigger.

**Syntax** :TRIGger:ENHanced:I2Cbus:GCALl:  
BIT7maddress?  
<x> = 1 or 2

**Example** :TRIGGER:ENHANCED:I2CBUS:GCALL:  
BIT7MADDRESS? -> :TRIGGER:ENHANCED:  
I2CBUS:GCALL:BIT7MADDRESS:  
PATTERN "1010101"

## 5.29 TRIGger Group

### **:TRIGger:ENHanced:I2CBus:GCALl:BIT7maddress:HEXA**

**Function** Sets the 7-bit master address of the general call of the I<sup>2</sup>C bus trigger in hexadecimal notation.

**Syntax** :TRIGger:ENHanced:I2CBus:GCALl:BIT7maddress:HEXA {<String>}  
<x> = 1 or 2  
<String> = 2 characters by combining '0' to 'F' and 'X' (bit 0 is fixed 1)

**Example** :TRIGGER:ENHANCED:I2CBUS:GCALL:BIT7MADDRESS:HEXA "AB"

### **:TRIGger:ENHanced:I2CBus:GCALl:BIT7maddress:PATtern**

**Function** Sets the 7-bit master address of the general call of the I<sup>2</sup>C bus trigger in binary notation or queries the current setting.

**Syntax** :TRIGger:ENHanced:I2CBus:GCALl:BIT7maddress:PATtern {<String>}  
:TRIGger:ENHanced:I2CBus:GCALl:BIT7maddress:PATtern?  
<x> = 1 or 2  
<String> = 7 characters by combining '0','1,' and 'X'

**Example** :TRIGGER:ENHANCED:I2CBUS:GCALL:BIT7MADDRESS:PATTERN "1010101"  
:TRIGGER:ENHANCED:I2CBUS:GCALL:BIT7MADDRESS:PATTERN? -> :TRIGGER:ENHANCED:I2CBUS:GCALL:BIT7MADDRESS:PATTERN "1010101"

### **:TRIGger:ENHanced:I2CBus:GCALl:SBYTE (Second Byte)**

**Function** Sets the second byte type of the general call of the I<sup>2</sup>C bus trigger or queries the current setting.

**Syntax** :TRIGger:ENHanced:I2CBus:GCALl:SBYTE {BIT7maddress|DONTcare|H04|H06}  
:TRIGger:ENHanced:I2CBus:GCALl:SBYTE?

**Example** :TRIGGER:ENHANCED:I2CBUS:GCALL:SBYTE BIT7MADDRESS  
:TRIGGER:ENHANCED:I2CBUS:GCALL:SBYTE?  
-> :TRIGGER:ENHANCED:I2CBUS:GCALL:SBYTE BIT7MADDRESS

### **:TRIGger:ENHanced:I2CBus:MODE**

**Function** Sets the trigger mode of the I<sup>2</sup>C bus trigger or queries the current setting.

**Syntax** :TRIGger:ENHanced:I2CBus:MODE {ADATa|ESTart|GCALl|NAIGNore|SBHSmode}  
:TRIGger:ENHanced:I2CBus:MODE?

**Example** :TRIGGER:ENHANCED:I2CBUS:MODE ADATA  
:TRIGGER:ENHANCED:I2CBUS:MODE?  
-> :TRIGGER:ENHANCED:I2CBUS:MODE ADATA

### **:TRIGger:ENHanced:I2CBus:NAIGNore?**

**Function** Queries all settings related to the NON ACK ignore mode of the I<sup>2</sup>C bus trigger.

**Syntax** :TRIGger:ENHanced:I2CBus:NAIGNore?

**Example** :TRIGGER:ENHANCED:I2CBUS:NAIGNORE?  
-> :TRIGGER:ENHANCED:I2CBUS:NAIGNORE:HSMODE 1;RACCESS 1;SBYTE 1

### **:TRIGger:ENHanced:I2CBus:NAIGNore:HSMode**

**Function** Sets whether to ignore NON ACK in high speed mode of the I<sup>2</sup>C bus trigger or queries the current setting.

**Syntax** :TRIGger:ENHanced:I2CBus:NAIGNore:HSMode {<Boolean>}  
:TRIGger:ENHanced:I2CBus:NAIGNore:HSMode?

**Example** :TRIGGER:ENHANCED:I2CBUS:NAIGNORE:HSMODE ON  
:TRIGGER:ENHANCED:I2CBUS:NAIGNORE:HSMODE? -> :TRIGGER:ENHANCED:I2CBUS:NAIGNORE:HSMODE 1

### **:TRIGger:ENHanced:I2CBus:NAIGNore:RACcEss**

**Function** Sets whether to ignore NON ACK in read access mode of the I<sup>2</sup>C bus trigger or queries the current setting.

**Syntax** :TRIGger:ENHanced:I2CBus:NAIGNore:RACcEss {<Boolean>}  
:TRIGger:ENHanced:I2CBus:NAIGNore:RACcEss?

**Example** :TRIGGER:ENHANCED:I2CBUS:NAIGNORE:RACCESS ON  
:TRIGGER:ENHANCED:I2CBUS:NAIGNORE:RACCESS? -> :TRIGGER:ENHANCED:I2CBUS:NAIGNORE:RACCESS 1

### **:TRIGger:ENHanced:I2CBus:NAIGNore:SBYTE (Start Byte)**

**Function** Sets whether to ignore NON ACK in the start byte of the I<sup>2</sup>C bus trigger or queries the current setting.

**Syntax** :TRIGger:ENHanced:I2CBus:NAIGNore:SBYTE {<Boolean>}  
:TRIGger:ENHanced:I2CBus:NAIGNore:SBYTE?

**Example** :TRIGGER:ENHANCED:I2CBUS:NAIGNORE:SBYTE ON  
:TRIGGER:ENHANCED:I2CBUS:NAIGNORE:SBYTE? -> :TRIGGER:ENHANCED:I2CBUS:NAIGNORE:SBYTE 1

**:TRIGger:ENHanced:I2CBus:SBHSmode?**

**Function** Queries all settings related to the start byte and high speed mode of the I<sup>2</sup>C bus trigger.

**Syntax** :TRIGger:ENHanced:I2CBus:SBHSmode?

**Example** :TRIGGER:ENHANCED:I2CBUS:SBHSMODE?  
-> :TRIGGER:ENHANCED:I2CBUS:SBHSMODE:  
TYPE HSMODE

**:TRIGger:ENHanced:I2CBus:SBHSmode:TYPE**

**Function** Sets the type of the start byte or high speed mode of the I<sup>2</sup>C bus trigger or queries the current setting.

**Syntax** :TRIGger:ENHanced:I2CBus:SBHSmode:  
TYPE {HSMODE|SBYTE}

**Example** :TRIGger:ENHanced:I2CBus:SBHSmode:TYPE?  
:TRIGGER:ENHANCED:I2CBUS:SBHSMODE:TYPE?  
-> :TRIGGER:ENHANCED:I2CBUS:SBHSMODE:  
TYPE HSMODE

**:TRIGger:ENHanced:LINBus?**

**Function** Queries all settings related to the LIN bus trigger or queries the current setting.

**Syntax** :TRIGger:ENHanced:LINBus?

**Example** :TRIGGER:ENHANCED:LINBUS? -> :TRIGGER:  
ENHANCED:LINBUS:BRATE 19200;  
SOURCE 1

**:TRIGger:ENHanced:LINBus:BRATE**

**Function** Sets the LIN bus signal trigger bitrate (data transfer rate) or queries the current setting.

**Syntax** :TRIGger:ENHanced:LINBus:BRATE {<NRf>|  
USER,<NRf>}  
:TRIGger:ENHanced:LINBus:BRATE?

**Example** <NRf> = 1200, 2400, 4800, 9600, 19200  
:TRIGGER:ENHANCED:LINBUS:BRATE 19200  
:TRIGGER:ENHANCED:LINBUS:BRATE?  
-> :TRIGGER:ENHANCED:LINBUS:BRATE 19200

**:TRIGger:ENHanced:LINBus:SOURce**

**Function** Sets the LIN bus signal trigger source or queries the current setting.

**Syntax** :TRIGger:ENHanced:LINBus:SOURce {<NRf>}  
:TRIGger:ENHanced:LINBus:SOURce?  
<NRf> = 1 to 4

**Example** :TRIGGER:ENHANCED:LINBUS:SOURCE 1  
:TRIGGER:ENHANCED:LINBUS:SOURCE?  
-> :TRIGGER:ENHANCED:LINBUS:SOURCE 1

**:TRIGger:ENHanced:SPATtern? (Serial Pattern)**

**Function** Queries all settings related to the serial pattern trigger.

**Syntax** :TRIGger:ENHanced:SPATtern?

**Example** :TRIGGER:ENHANCED:SPATTERN?  
-> :TRIGGER:ENHANCED:SPATTERN:  
BITRATE 1.000E+00;CLOCK:MODE 1;  
POLARITY FALL;SOURCE 1;:TRIGGER:  
ENHANCED:SPATTERN:CS 1;DATA:  
ACTIVE HIGH;SOURCE 1;:TRIGGER:ENHANCED:  
SPATTERN:LATCH:SOURCE 1;POLARITY FALL;:  
TRIGGER:ENHANCED:SPATTERN:  
PATTERN "1100110111101111"

**:TRIGger:ENHanced:SPATtern:BITRate**

**Function** Sets the bit rate of the serial pattern trigger or queries the current setting.

**Syntax** :TRIGger:ENHanced:SPATtern:  
BITRate {<NRf>}  
:TRIGger:ENHanced:SPATtern:BITRate?  
<NRf> = 1 to 50M (bps)

**Example** :TRIGGER:ENHANCED:SPATTERN:BITRATE 1  
:TRIGGER:ENHANCED:SPATTERN:BITRATE?  
-> :TRIGGER:ENHANCED:SPATTERN:  
BITRATE 1.000E+00

**Description** This command is valid when :TRIGger:ENHanced:  
SPATtern:CLOCK:MODE OFF.

**:TRIGger:ENHanced:SPATtern:CLEAr**

**Function** Clears the entire pattern of the serial pattern trigger (to don't care).

**Syntax** :TRIGger:ENHanced:SPATtern:CLEAr  
**Example** :TRIGGER:ENHANCED:SPATTERN:CLEAR

**:TRIGger:ENHanced:SPATtern:CLOCK?**

**Function** Queries all settings related to clock of the serial pattern trigger.

**Syntax** :TRIGger:ENHanced:SPATtern:CLOCK?  
**Example** :TRIGGER:ENHANCED:SPATTERN:CLOCK?  
-> :TRIGGER:ENHANCED:SPATTERN:CLOCK:  
MODE 1;POLARITY FALL;SOURCE 1

**:TRIGger:ENHanced:SPATtern:CLOCK:MODE**

**Function** Enables/Disables the clock of the serial pattern trigger or queries the current setting.

**Syntax** :TRIGger:ENHanced:SPATtern:CLOCK:  
MODE {<Boolean>}  
:TRIGger:ENHanced:SPATtern:CLOCK:MODE?  
**Example** :TRIGGER:ENHANCED:SPATTERN:CLOCK:  
MODE ON  
:TRIGGER:ENHANCED:SPATTERN:CLOCK:MODE?  
-> :TRIGGER:ENHANCED:SPATTERN:CLOCK:  
MODE 1

## 5.29 TRIGger Group

### **:TRIGger:ENHanced:SPATtern:CLOCK:POLARity**

**Function** Sets the polarity of the clock trace of the serial pattern trigger or queries the current setting.

**Syntax** :TRIGger:ENHanced:SPATtern:CLOCK:POLARity {FALL|RISE}  
:TRIGger:ENHanced:SPATtern:CLOCK:POLARity?

**Example** :TRIGGER:ENHANCED:SPATTERN:CLOCK:POLARITY FALL  
:TRIGGER:ENHANCED:SPATTERN:CLOCK:POLARITY? -> :TRIGGER:ENHANCED:SPATTERN:CLOCK:POLARITY FALL

**Description** This command is valid when :TRIGger:ENHanced:SPATtern:CLOCK:MODE ON.

### **:TRIGger:ENHanced:SPATtern:CLOCK:SOURce**

**Function** Sets the clock trace of the serial pattern trigger or queries the current setting.

**Syntax** :TRIGger:ENHanced:SPATtern:CLOCK:SOURce {<Nrf>}  
:TRIGger:ENHanced:SPATtern:CLOCK:SOURce?  
<Nrf> = 1 to 4

**Example** :TRIGGER:ENHANCED:SPATTERN:CLOCK:SOURCE 1  
:TRIGGER:ENHANCED:SPATTERN:CLOCK:SOURCE? -> :TRIGGER:ENHANCED:SPATTERN:CLOCK:SOURCE 1

**Description** This command is valid when :TRIGger:ENHanced:SPATtern:CLOCK:MODE ON.

### **:TRIGger:ENHanced:SPATtern:CS**

**Function** Enables/Disables the chip select of the serial pattern trigger or queries the current setting.

**Syntax** :TRIGger:ENHanced:SPATtern:CS {<Boolean>}  
:TRIGger:ENHanced:SPATtern:CS?

**Example** :TRIGGER:ENHANCED:SPATTERN:CS ON  
:TRIGGER:ENHANCED:SPATTERN:CS?  
-> :TRIGGER:ENHANCED:SPATTERN:CS 1

**Description** This command is valid when :TRIGger:ENHanced:SPATtern:CLOCK:MODE ON.

### **:TRIGger:ENHanced:SPATtern:DATA?**

**Function** Queries all settings related to data of the serial pattern trigger.

**Syntax** :TRIGger:ENHanced:SPATtern:DATA?

**Example** :TRIGGER:ENHANCED:SPATTERN:DATA?  
-> :TRIGGER:ENHANCED:SPATTERN:DATA:ACTIVE HIGH;SOURCE 1

### **:TRIGger:ENHanced:SPATtern:DATA:ACTive**

**Function** Sets the active level of the data of the serial pattern trigger or queries the current setting.

**Syntax** :TRIGger:ENHanced:SPATtern:DATA:ACTive {HIGH|LOW}  
:TRIGger:ENHanced:SPATtern:DATA:ACTive?

**Example** :TRIGGER:ENHANCED:SPATTERN:DATA:ACTIVE HIGH  
:TRIGGER:ENHANCED:SPATTERN:DATA:ACTIVE?  
-> :TRIGGER:ENHANCED:SPATTERN:DATA:ACTIVE HIGH

### **:TRIGger:ENHanced:SPATtern:DATA:SOURce**

**Function** Sets the data trace of the serial pattern trigger or queries the current setting.

**Syntax** :TRIGger:ENHanced:SPATtern:DATA:SOURce {<Nrf>}  
:TRIGger:ENHanced:SPATtern:DATA:SOURce?<Nrf> = 1 to 4

**Example** :TRIGGER:ENHANCED:SPATTERN:DATA:SOURCE 1  
:TRIGGER:ENHANCED:SPATTERN:DATA:SOURCE?  
-> :TRIGGER:ENHANCED:SPATTERN:DATA:SOURCE 1

### **:TRIGger:ENHanced:SPATtern:HEXA**

**Function** Sets the pattern of the serial pattern trigger in hexadecimal notation.

**Syntax** :TRIGger:ENHanced:SPATtern:HEXA {<String>}

<String> = Up to 32 characters by combining '0' to 'F' and 'X'

**Example** :TRIGGER:ENHANCED:SPATTERN:HEXA "ABCD"

### **:TRIGger:ENHanced:SPATtern:LATCh?**

**Function** Queries all settings related to latch of the serial pattern trigger.

**Syntax** :TRIGger:ENHanced:SPATtern:LATCh?

**Example** :TRIGGER:ENHANCED:SPATTERN:LATCH?  
-> :TRIGGER:ENHANCED:SPATTERN:LATCH:SOURCE 1;POLARITY FALL



**:TRIGger:ENHanced:SPATtern:LATCh:****POLarity**

**Function** Sets the polarity of the latch trace of the serial pattern trigger or queries the current setting.

**Syntax** :TRIGger:ENHanced:SPATtern:LATCh:  
POLarity {FALL|RISE}  
:TRIGger:ENHanced:SPATtern:LATCh:  
POLarity?

**Example** :TRIGGER:ENHANCED:SPATTERN:LATCH:  
POLARITY FALL  
:TRIGGER:ENHANCED:SPATTERN:LATCH:  
POLARITY? -> :TRIGGER:ENHANCED:  
SPATTERN:LATCH:POLARITY FALL

**Description** • This command is valid when :TRIGger:  
ENHanced:SPATtern:CLOCK:MODE ON.  
• This command is invalid if :TRIGger:ENHanced:  
SPATtern:LATCh:SOURce NONE.

**:TRIGger:ENHanced:SPATtern:LATCh:****SOURce**

**Function** Sets the latch trace of the serial pattern trigger or queries the current setting.

**Syntax** :TRIGger:ENHanced:SPATtern:LATCh:  
SOURce {<NRF>|NONE}  
:TRIGger:ENHanced:SPATtern:LATCh:  
SOURce?  
<NRF> = 1 to 4

**Example** :TRIGGER:ENHANCED:SPATTERN:LATCH:  
SOURCE 1  
:TRIGGER:ENHANCED:SPATTERN:LATCH:  
SOURCE? -> :TRIGGER:ENHANCED:SPATTERN:  
LATCH:SOURCE 1

**Description** This command is valid when :TRIGger:ENHanced:  
SPATtern:CLOCK:MODE ON.

**:TRIGger:ENHanced:SPATtern:PATtern**

**Function** Sets the pattern of the serial pattern trigger in binary notation or queries the current setting.

**Syntax** :TRIGger:ENHanced:SPATtern:  
PATtern {<String>}  
:TRIGger:ENHanced:SPATtern:PATtern?  
<String> = Up to 128 characters by combining '0', '1',  
and 'X'

**Example** :TRIGGER:ENHANCED:SPATTERN:PATTERN  
"1100110111101111"  
:TRIGGER:ENHANCED:SPATTERN:PATTERN?  
-> :TRIGGER:ENHANCED:SPATTERN:  
PATTERN "1100110111101111"

**:TRIGger:ENHanced:SPIBus?**

**Function** Queries all settings related to the SPI bus trigger.

**Syntax** :TRIGger:ENHanced:SPIBus?  
**Example** :TRIGGER:ENHANCED:SPIBUS? -> :TRIGGER:  
ENHANCED:SPIBUS:BITORDER LSBFIRST;  
CLOCK:POLARITY FALL;SOURCE 1;:TRIGGER:  
ENHANCED:SPIBUS:CS:ACTIVE HIGH;  
SOURCE 1;:TRIGGER:ENHANCED:SPIBUS:  
DATA1:BYTE 1;CONDITION TRUE;  
DPOSITION 1;PATTERN1 "00010010";  
PATTERN2 "00110100";  
PATTERN3 "01010110";  
PATTERN4 "00010010";SOURCE 3;:TRIGGER:  
ENHANCED:SPIBUS:DATA2:BYTE 4;  
CONDITION TRUE;DPOSITION 1;  
PATTERN1 "00010010";  
PATTERN2 "00110100";  
PATTERN3 "01010110";  
PATTERN4 "00010010";SOURCE 3;:TRIGGER:  
ENHANCED:SPIBUS:MODE WIRE3

**:TRIGger:ENHanced:SPIBus:BITorder**

**Function** Sets the bit order of the SPI bus trigger or queries the current setting.

**Syntax** :TRIGger:ENHanced:SPIBus:  
BITorder {LSBFirst|MSBFirst}  
:TRIGger:ENHanced:SPIBus:BITorder?

**Example** :TRIGGER:ENHANCED:SPIBUS:  
BITORDER LSBFIRST  
:TRIGGER:ENHANCED:SPIBUS:BITORDER?  
-> :TRIGGER:ENHANCED:SPIBUS:  
BITORDER LSBFIRST

**:TRIGger:ENHanced:SPIBus:CLOCK?**

**Function** Queries all settings related to the clock of the SPI bus trigger.

**Syntax** :TRIGger:ENHanced:SPIBus:CLOCK?  
**Example** :TRIGGER:ENHANCED:SPIBUS:CLOCK?  
-> :TRIGGER:ENHANCED:SPIBUS:CLOCK:  
POLARITY FALL;SOURCE 1

**:TRIGger:ENHanced:SPIBus:CLOCK:****POLarity**

**Function** Sets the polarity of the clock trace of the SPI bus trigger or queries the current setting.

**Syntax** :TRIGger:ENHanced:SPIBus:CLOCK:  
POLarity {FALL|RISE}  
:TRIGger:ENHanced:SPIBus:CLOCK:  
POLarity?

**Example** :TRIGGER:ENHANCED:SPIBUS:CLOCK:  
POLARITY FALL  
:TRIGGER:ENHANCED:SPIBUS:CLOCK:  
POLARITY? -> :TRIGGER:ENHANCED:SPIBUS:  
CLOCK:POLARITY FALL

## 5.29 TRIGger Group

**:TRIGger:ENHanced:SPIBus:CLOCK:SOURce**  
Function Sets the clock trace of the SPI bus trigger or queries the current setting.

Syntax :TRIGger:ENHanced:SPIBus:CLOCK:  
SOURce {<NRf>}  
:TRIGger:ENHanced:SPIBus:CLOCK:SOURce?  
<NRf> = 1 to 4

Example :TRIGGER:ENHANCED:SPIBUS:CLOCK:SOURCE 1  
:TRIGGER:ENHANCED:SPIBUS:CLOCK:SOURCE?  
-> :TRIGGER:ENHANCED:SPIBUS:CLOCK:  
SOURCE 1

### **:TRIGger:ENHanced:SPIBus:CS?**

Function Queries all settings related to the chip select of the SPI bus trigger.

Syntax :TRIGger:ENHanced:SPIBus:CS?

Example :TRIGGER:ENHANCED:SPIBUS:CS?  
-> :TRIGGER:ENHANCED:SPIBUS:CS:  
ACTIVE HIGH;SOURCE 1

### **:TRIGger:ENHanced:SPIBus:CS:ACTive**

Function Sets the active level of the chip select of the SPI bus trigger or queries the current setting.

Syntax :TRIGger:ENHanced:SPIBus:CS:  
ACTive {HIGH|LOW}  
:TRIGger:ENHanced:SPIBus:CS:ACTive?

Example :TRIGGER:ENHANCED:SPIBUS:CS:ACTIVE HIGH  
:TRIGGER:ENHANCED:SPIBUS:CS:ACTIVE?  
-> :TRIGGER:ENHANCED:SPIBUS:CS:  
ACTIVE HIGH

### **:TRIGger:ENHanced:SPIBus:CS:SOURce**

Function Sets the chip select trace of the SPI bus trigger or queries the current setting.

Syntax :TRIGger:ENHanced:SPIBus:CS:  
SOURce {<NRf>}  
:TRIGger:ENHanced:SPIBus:CS:SOURce?  
<NRf> = 1 to 4

Example :TRIGGER:ENHANCED:SPIBUS:CS:SOURCE 1  
:TRIGGER:ENHANCED:SPIBUS:CS:SOURCE?  
-> :TRIGGER:ENHANCED:SPIBUS:CS:SOURCE 1

### **:TRIGger:ENHanced:SPIBus:DATA<x>?**

Function Queries all settings related to the data of the SPI bus trigger.

Syntax :TRIGger:ENHanced:SPIBus:DATA<x>?  
<x> = 1 or 2

Example :TRIGGER:ENHANCED:SPIBUS:DATA1? -> :  
TRIGGER:ENHANCED:SPIBUS:DATA1:BYTE 1;  
CONDITION TRUE;DPOSITION 1;  
PATTERN1 "00010010";  
PATTERN2 "00110100";  
PATTERN3 "01010110";  
PATTERN4 "00010010";SOURCE 3

Description DATA2 is valid when :TRIGger:ENHanced:  
SPIBus:MODE WIRE4 is specified.

**:TRIGger:ENHanced:SPIBus:DATA<x>:BYTE**  
Function Sets the number of bytes of the data of the SPI bus trigger or queries the current setting.

Syntax :TRIGger:ENHanced:SPIBus:DATA<x>:  
BYTE {<NRf>}  
:TRIGger:ENHanced:SPIBus:DATA<x>:BYTE?  
<x> = 1 or 2  
<NRf> = 1 to 4

Example :TRIGGER:ENHANCED:SPIBUS:DATA1:BYTE 1  
:TRIGGER:ENHANCED:SPIBUS:DATA1:BYTE?  
-> :TRIGGER:ENHANCED:SPIBUS:DATA1:  
BYTE 1

### **:TRIGger:ENHanced:SPIBus:DATA<x>: CONDition**

Function Sets the determination method (match or not match) of the data of the SPI bus trigger or queries the current setting.

Syntax :TRIGger:ENHanced:SPIBus:DATA<x>:  
CONDition {FALSE|TRUE}  
:TRIGger:ENHanced:SPIBus:DATA<x>:  
CONDition?  
<x> = 1 or 2

Example :TRIGGER:ENHANCED:SPIBUS:DATA1:  
CONDITION TRUE  
:TRIGGER:ENHANCED:SPIBUS:DATA1:  
CONDITION? -> :TRIGGER:ENHANCED:SPIBUS:  
DATA1:CONDITION TRUE

### **:TRIGger:ENHanced:SPIBus:DATA<x>: DPOSITion**

Function Sets the pattern comparison start position of the data of the SPI bus trigger or queries the current setting.

Syntax :TRIGger:ENHanced:SPIBus:DATA<x>:  
DPOSITion {<NRf>}  
:TRIGger:ENHanced:SPIBus:DATA<x>:  
DPOSITion?  
<x> = 1 or 2  
<NRf> = 0 to 9999

Example :TRIGGER:ENHANCED:SPIBUS:DATA1:  
DPOSITION 1  
:TRIGGER:ENHANCED:SPIBUS:DATA1:  
DPOSITION? -> :TRIGGER:ENHANCED:SPIBUS:  
DATA1:DPOSITION 1

**:TRIGger:ENHanced:SPIBus:DATA<x>:HEXA<x>**

**Function** Sets the data of the SPI bus trigger in hexadecimal notation.

**Syntax** :TRIGger:ENHanced:SPIBus:DATA<x>:  
HEXA<x> {<String>}  
<x> of DATA<x> = 1 or 2  
<x> of HEXA<x> = 1 to 4  
<String> = 2 characters by combining '0' to 'F' and 'X'

**Example** :TRIGGER:ENHANCED:SPIBUS:DATA1:  
HEXA1 "AB"

**:TRIGger:ENHanced:SPIBus:DATA<x>:PATTern<x>**

**Function** Sets the data of the SPI bus trigger in binary notation or queries the current setting.

**Syntax** :TRIGger:ENHanced:SPIBus:DATA<x>:  
PATTern<x> {<String>}  
:TRIGger:ENHanced:SPIBus:DATA<x>:  
PATTern<x>?  
<x> of DATA<x> = 1 or 2  
<x> of PATTern<x> = 1 to 4  
<String> = 8 characters by combining '0', '1,' and 'X'

**Example** :TRIGGER:ENHANCED:SPIBUS:DATA1:  
PATTERN1 "10101011"  
:TRIGGER:ENHANCED:SPIBUS:DATA1:  
PATTERN1? -> :TRIGGER:ENHANCED:SPIBUS:  
DATA1:PATTERN1 "10101011"

**:TRIGger:ENHanced:SPIBus:DATA<x>:SOURce**

**Function** Sets the trace of the data of the SPI bus trigger or queries the current setting.

**Syntax** :TRIGger:ENHanced:SPIBus:DATA<x>:  
SOURce {<NRf>}  
:TRIGger:ENHanced:SPIBus:DATA<x>:  
SOURce?  
<x> = 1 or 2  
<NRf> = 1 to 4

**Example** :TRIGGER:ENHANCED:SPIBUS:DATA1:SOURCE 1  
:TRIGGER:ENHANCED:SPIBUS:DATA1:SOURCE?  
-> :TRIGGER:ENHANCED:SPIBUS:DATA1:  
SOURCE 1

**:TRIGger:ENHanced:SPIBus:MODE**

**Function** Sets the wiring system of the SPI bus trigger (three-wire or four-wire) or queries the current setting.

**Syntax** :TRIGger:ENHanced:SPIBus:MODE {WIRE3|  
WIRE4}  
:TRIGger:ENHanced:SPIBus:MODE?

**Example** :TRIGGER:ENHANCED:SPIBUS:MODE WIRE3  
:TRIGGER:ENHANCED:SPIBUS:MODE?  
-> :TRIGGER:ENHANCED:SPIBUS:MODE WIRE3

**:TRIGger:ENHanced:TV?**

**Function** Queries all settings related to the TV trigger.

**Syntax** :TRIGger:ENHanced:TV?

**Example** :TRIGGER:ENHANCED:TV?  
-> :TRIGGER:ENHANCED:TV:CUSTOMIZE 1;  
FIELD DONTCARE;FRAME 2;HDTV:LINE 2;  
POLARITY NEGATIVE;;TRIGGER:ENHANCED:  
TV:LEVEL 1.000E+00;NTSC:LINE 5;  
POLARITY NEGATIVE;;TRIGGER:ENHANCED:TV:  
PAL:LINE 2;POLARITY NEGATIVE;;TRIGGER:  
ENHANCED:TV:SDTV:LINE 8;  
POLARITY NEGATIVE;;TRIGGER:ENHANCED:TV:  
SGUARD 60;SOURCE 1;TYPE HDTV;  
USERDEFINE:DEFINITION HD;  
HFREJECTION OFF;HSYNC 50.00E+06;LINE 2;  
POLARITY NEGATIVE

**:TRIGger:ENHanced:TV:COUPling?**

**Function** Queries the trigger coupling of the TV trigger.

**Syntax** :TRIGger:ENHanced:TV:COUPling?

**Example** :TRIGGER:ENHANCED:TV:COUPLING?  
-> :TRIGGER:ENHANCED:TV:COUPLING TV

**:TRIGger:ENHanced:TV:CUSTomize**

**Function** Turns ON/OFF the sync guard function of the TV trigger or queries the current setting.

**Syntax** :TRIGger:ENHanced:TV:  
CUSTomize {<Boolean>}

:TRIGger:ENHanced:TV:CUSTomize?  
**Example** :TRIGGER:ENHANCED:TV:CUSTOMIZE ON  
:TRIGGER:ENHANCED:TV:CUSTOMIZE?  
-> :TRIGGER:ENHANCED:TV:CUSTOMIZE 1

**:TRIGger:ENHanced:TV:FIELD**

**Function** Sets the field of the TV trigger or queries the current setting.

**Syntax** :TRIGger:ENHanced:TV:FIELD {DONTcare|  
<NRf>}  
:TRIGger:ENHanced:TV:FIELD?  
<NRf> = 1 or 2

**Example** :TRIGGER:ENHANCED:TV:FIELD DONTCARE  
:TRIGGER:ENHANCED:TV:FIELD?  
-> :TRIGGER:ENHANCED:TV:FIELD DONTCARE

**:TRIGger:ENHanced:TV:FRAME**

**Function** Sets the frame skip function of the TV trigger or queries the current setting.

**Syntax** :TRIGger:ENHanced:TV:FRAME {<NRf>}  
:TRIGger:ENHanced:TV:FRAME?  
<NRf> = 1, 2, 4, or 8

**Example** :TRIGGER:ENHANCED:TV:FRAME 2  
:TRIGGER:ENHANCED:TV:FRAME?  
-> :TRIGGER:ENHANCED:TV:FRAME 2

## 5.29 TRIGger Group

### **:TRIGger:ENHanced:TV:{HDTV|NTSC|PAL|SDTV|USERdefine}?**

Function Queries all settings related to the TV trigger mode.

Syntax :TRIGger:ENHanced:TV:{HDTV|NTSC|PAL|SDTV|USERdefine}?

Example (The following is an example for the HDTV.)

```
:TRIGGER:ENHANCED:TV:HDTV?
-> :TRIGGER:ENHANCED:TV:HDTV:LINE 2;
POLARITY NEGATIVE
```

### **:TRIGger:ENHanced:TV:{HDTV|NTSC|PAL|SDTV}:HFRejection?** **(HighFrequencyREJECTION)**

Function Queries the low pass filter (HF rejection) of the TV trigger.

Syntax TRIGger:ENHanced:TV:{HDTV|NTSC|PAL|SDTV}:HFRejection?

Example (The following is an example for the HDTV.)

```
:TRIGGER:ENHANCED:TV:HDTV:HFREJECTION?
-> :TRIGGER:ENHANCED:TV:HDTV:
HFREJECTION OFF
```

### **:TRIGger:ENHanced:TV:{HDTV|NTSC|PAL|SDTV|USERdefine}:LINE**

Function Sets the line for activating the TV trigger or queries the current setting.

Syntax :TRIGger:ENHanced:TV:{HDTV|NTSC|PAL|SDTV|USERdefine}:LINE  
:TRIGger:ENHanced:TV:{HDTV|NTSC|PAL|SDTV|USERdefine}:LINE {<Nrf>}  
<Nrf> = 2 to 2251 (for HDTV)  
5 to 1054 (for NTSC)  
2 to 1251 (for PAL)  
8 to 2251 (for SDTV)  
2 to 2251 (for USERdefine)

Example (The following is an example for the HDTV.)

```
:TRIGGER:ENHANCED:TV:HDTV:LINE 10
:TRIGGER:ENHANCED:TV:HDTV:LINE?
-> :TRIGGER:ENHANCED:TV:HDTV:LINE 10
```

### **:TRIGger:ENHanced:TV:{HDTV|NTSC|PAL|SDTV|USERdefine}:POLarity**

Function Sets the input polarity of the TV trigger or queries the current setting.

Syntax :TRIGger:ENHanced:TV:{HDTV|NTSC|PAL|SDTV|USERdefine}:POLarity  
{NEGative|POSitive}  
:TRIGger:ENHanced:TV:{HDTV|NTSC|PAL|SDTV|USERdefine}:POLarity?

Example (The following is an example for the HDTV.)

```
:TRIGGER:ENHANCED:TV:HDTV:
POLARITY NEGATIVE
:TRIGGER:ENHANCED:TV:HDTV:POLARITY?
-> :TRIGGER:ENHANCED:TV:HDTV:
POLARITY NEGATIVE
```

### **:TRIGger:ENHanced:TV:LEVel**

Function Sets the trigger level of the TV trigger or queries the current setting.

Syntax :TRIGger:ENHanced:TV:LEVel {<Nrf>}  
:TRIGger:ENHanced:TV:LEVel?  
<Nrf> = 0.1 to 2.0 (div)

Example :TRIGGER:ENHANCED:TV:LEVEL 1

```
:TRIGGER:ENHANCED:TV:LEVEL?
-> :TRIGGER:ENHANCED:TV:LEVEL 1.000E+00
```

### **:TRIGger:ENHanced:TV:SGUard**

Function Sets the sync guard of the TV trigger or queries the current setting.

Syntax :TRIGger:ENHanced:TV:SGUard {<Nrf>}  
:TRIGger:ENHanced:TV:SGUard?  
<Nrf> = 60 to 90 (%)

Example :TRIGGER:ENHANCED:TV:SGUARD 60

```
:TRIGGER:ENHANCED:TV:SGUARD?
-> :TRIGGER:ENHANCED:TV:SGUARD 60
```

Description This command is valid when :TRIGGER:ENHANCED:TV:TYPE HDTV|NTSC|PAL.

### **:TRIGger:ENHanced:TV:SOURce**

Function Sets the trigger source of the TV trigger or queries the current setting.

Syntax :TRIGger:ENHanced:TV:SOURce {<Nrf>}  
:TRIGger:ENHanced:TV:SOURce?  
<Nrf> = 1 to 4

Example :TRIGGER:ENHANCED:TV:SOURCE 1

```
:TRIGGER:ENHANCED:TV:SOURCE?
-> :TRIGGER:ENHANCED:TV:SOURCE 1
```

### **:TRIGger:ENHanced:TV:TYPE**

Function Sets the input type of the TV trigger or queries the current setting.

Syntax :TRIGger:ENHanced:TV:TYPE {HDTV|NTSC|PAL|SDTV|USERdefine,I1080\_50|I1080\_60|P1080\_24|P1080\_25|P1080\_60|P720\_60|SF1080\_24}  
:TRIGger:ENHanced:TV:TYPE?

Example :TRIGGER:ENHANCED:TV:TYPE NTSC

```
:TRIGGER:ENHANCED:TV:TYPE? -> TRIGGER:
ENHANCED:TV:TYPE NTSC
```

Description For {HDTV}, select the next {I1080\_50|I1080\_60|P1080\_24|P1080\_25|P1080\_60|P720\_60|SF1080\_24}. If not selected {I1080\_60} is selected automatically.

**:TRIGger:ENHanced:TV:USERdefine:****DEFinition**

**Function** Sets the user-defined resolution or queries the current setting.

**Syntax** :TRIGger:ENHanced:TV:USERdefine:  
DEFinition {HD|SD}  
:TRIGger:ENHanced:TV:USERdefine:  
DEFinition?

**Example** :TRIGGER:ENHANCED:TV:USERDEFINE:  
DEFINITION HD  
:TRIGGER:ENHANCED:TV:USERDEFINE:  
DEFINITION? -> :TRIGGER:ENHANCED:TV:  
USERDEFINE:DEFINITION HD

**:TRIGger:ENHanced:TV:USERdefine:****HFRejection (HighFrequencyREJECTION)**

**Function** Sets the user-defined low pass filter (HF rejection) or queries the current setting.

**Syntax** :TRIGger:ENHanced:TV:USERdefine:  
HFRejection {<Frequency>|OFF}  
:TRIGger:ENHanced:TV:USERdefine:  
HFRejection?

**Example** :TRIGGER:ENHANCED:TV:USERDEFINE:  
HFREJECTION OFF  
:TRIGGER:ENHANCED:TV:USERDEFINE:  
HFREJECTION? -> :TRIGGER:ENHANCED:TV:  
USERDEFINE:HFREJECTION OFF

**:TRIGger:ENHanced:TV:USERdefine:****HSYnc (Hsync Freq)**

**Function** Sets the user-defined horizontal sync signal or queries the current setting.

**Syntax** :TRIGger:ENHanced:TV:USERdefine:  
HSYnc {<Frequency>}  
:TRIGger:ENHanced:TV:USERdefine:HSYnc?  
<Frequency> = 10k to 200k (Hz)

**Example** :TRIGGER:ENHANCED:TV:USERDEFINE:  
HSYNC 10KHZ  
:TRIGGER:ENHANCED:TV:USERDEFINE:HSYNC?  
-> :TRIGGER:ENHANCED:TV:USERDEFINE:  
HSYNC 10.00E+03

**:TRIGger:ENHanced:UART?**

**Function** Queries all settings related to the UART signal trigger.

**Syntax** :TRIGger:ENHanced:UART?

**Example** :TRIGGER:ENHANCED:UART? -> :TRIGGER:  
ENHANCED:UART:BRATE 19200;  
FORMAT BIT7PARITY;POLARITY NEGATIVE;  
SOURCE 1;SPOINT 18.8E+00

**:TRIGger:ENHanced:UART:BRATe**

**Function** Sets the UART signal trigger bit rate (data transfer rate) or queries the current setting.

**Syntax** :TRIGger:ENHanced:UART:  
BRATe {<Nrf>|USER, <Nrf>}  
:TRIGger:ENHanced:UART:BRATe?  
<Nrf> = 1200, 2400, 4800, 9600, 19200, 38400,  
57600, 115200  
<Nrf> of USER = See the DL9040/DL9140/DL9240  
User's Manual

**Example** :TRIGGER:ENHANCED:UART:BRATE 19200  
:TRIGGER:ENHANCED:UART:BRATE? ->  
:TRIGGER:ENHANCED:UART:BRATE 19200

**:TRIGger:ENHanced:UART:FORMat**

**Function** Sets the UART signal trigger format or queries the current setting.

**Syntax** :TRIGger:ENHanced:UART:  
FORMat {BIT7parity|BIT8Noparity|  
BIT8Parity}  
:TRIGger:ENHanced:UART:FORMat?

**Example** :TRIGGER:ENHANCED:UART:  
FORMAT BIT7PARITY  
:TRIGGER:ENHANCED:UART:FORMAT? ->  
:TRIGGER:ENHANCED:UART:  
FORMAT BIT7PARITY

**:TRIGger:ENHanced:UART:POLarity**

**Function** Sets the UART signal trigger polarity or queries the current setting.

**Syntax** :TRIGger:ENHanced:UART:  
POLarity {NEGative|POSitive}  
:TRIGger:ENHanced:UART:POLarity?

**Example** :TRIGGER:ENHANCED:UART:  
POLARITY NEGATIVE  
:TRIGGER:ENHANCED:UART:POLARITY? ->  
:TRIGGER:ENHANCED:UART:  
POLARITY NEGATIVE

**:TRIGger:ENHanced:UART:SOURce**

**Function** Sets the UART signal trigger source or queries the current setting.

**Syntax** :TRIGger:ENHanced:UART:SOURce {<Nrf>}  
:TRIGger:ENHanced:UART:SOURce?  
<Nrf> = 1 to 4

**Example** :TRIGGER:ENHANCED:UART:SOURCE 1  
:TRIGGER:ENHANCED:UART:SOURCE? ->  
:TRIGGER:ENHANCED:UART:SOURCE 1

**:TRIGger:ENHanced:UART:SPOint**

**Function** Sets the UART signal trigger sample point or queries the current setting.

**Syntax** :TRIGger:ENHanced:UART:SPOint {<Nrf>}  
:TRIGger:ENHanced:UART:SPOint?  
<Nrf> = 18.8 to 90.6(%)

**Example** :TRIGGER:ENHANCED:UART:SPOINT 18.8  
:TRIGGER:ENHANCED:UART:SPOINT? ->  
:TRIGGER:ENHANCED:UART:SPOINT 18.8E+00

## 5.29 TRIGger Group

### **:TRIGger:ESTate?**

**Function** Queries all settings related to the edge/state trigger.

**Syntax** :TRIGger:ESTate?

**Example** :TRIGGER:ESTATE? -> :TRIGGER:ESTATE:  
EOR:CHANNEL1 DONTCARE;  
CHANNEL2 DONTCARE;CHANNEL3 DONTCARE;  
CHANNEL4 DONTCARE;:TRIGGER:ESTATE:  
SOURCE 1;POLARITY ENTER

### **:TRIGger:ESTate:EOR?**

**Function** Queries all settings related to the OR trigger.

**Syntax** :TRIGger:ESTate:EOR?

**Example** :TRIGGER:ESTATE:EOR? -> :TRIGGER:  
ESTATE:EOR:CHANNEL1 DONTCARE;  
CHANNEL2 DONTCARE;CHANNEL3 DONTCARE;  
CHANNEL4 DONTCARE

### **:TRIGger:ESTate:EOR:CHANnel<x>**

**Function** Sets the channel polarity of the OR trigger or queries the current setting.

**Syntax** :TRIGger:ESTate:EOR:  
CHANnel<x> {DONTcare|ENTER|EXIT|FALL|  
RISE}  
:TRIGger:ESTate:EOR:CHANnel<x>?  
<x> = 1 to 4

**Example** :TRIGGER:ESTATE:EOR:CHANNEL1 DONTCARE  
:TRIGGER:ESTATE:EOR:CHANNEL1?  
-> :TRIGGER:ESTATE:EOR:  
CHANNEL1 DONTCARE

**Description** • This command is valid when :TRIGger:TYPE  
EOR.  
• {ENTER|EXIT} is valid when :TRIGger:SOURce:  
CHANnel<x>:WINDow ON. For all other cases,  
{FALL|RISE} is valid.

### **:TRIGger:ESTate:POLarity**

**Function** Sets the polarity of the edge/state trigger or queries the current setting.

**Syntax** :TRIGger:ESTate:POLarity {ENTER|EXIT|  
FALL|RISE}  
:TRIGger:ESTate:POLarity?

**Example** :TRIGGER:ESTATE:POLARITY ENTER  
:TRIGGER:ESTATE:POLARITY? -> :TRIGGER:  
ESTATE:POLARITY ENTER

**Description** • This command is valid when :TRIGger:TYPE  
EDGE|EQUalify|STATE.  
• This command is invalid when :TRIGger:TYPE  
EDGE and :TRIGger:ESTate:SOURce LINE.  
• {ENTER|EXIT} is valid when :TRIGger:TYPE  
EDGE|EQUalify and :TRIGger:SOURce:  
CHANnel<x>:WINDow ON. {FALL|RISE} is valid  
when :TRIGger:TYPE EDGE|EQUalify and :  
TRIGger:SOURce:CHANnel<x>:WINDow OFF.  
• {ENTER|EXIT} is valid when :TRIGger:TYPE  
STATE.

### **:TRIGger:ESTate:SOURce**

**Function** Sets the trigger source of the edge/state trigger or queries the current setting.

**Syntax** :TRIGger:ESTate:SOURce {<NRf>|EXTernal|  
LINE}  
:TRIGger:ESTate:SOURce?  
<NRf> = 1 to 4

**Example** :TRIGGER:ESTATE:SOURCE EXTERNAL  
:TRIGGER:ESTATE:SOURCE? -> :TRIGGER:  
ESTATE:SOURCE EXTERNAL

**Description** • This command is valid when :TRIGger:TYPE  
EDGE|EQUalify.  
• {<NRf>|EXTernal|LINE} is valid when :TRIGger:  
TYPE EDGE.  
• {<NRf>|EXTernal} is valid when :TRIGger:TYPE  
EQUalify.

### **:TRIGger:HOLDoff**

**Function** Sets the hold off time or queries the current setting.

**Syntax** :TRIGger:HOLDoff {<Time>}  
:TRIGger:HOLDoff?  
<Time> = 20 ns to 10 s in 5-ns steps

**Example** :TRIGGER:HOLDOFF 1S  
:TRIGGER:HOLDOFF?  
-> :TRIGGER:HOLDOFF 1.000E+00

### **:TRIGger:MODE**

**Function** Sets the trigger mode or queries the current setting.

**Syntax** :TRIGger:MODE {ALEvel|AUTO|NORMAL|  
NSINGLE|SINGLE}  
:TRIGger:MODE?

**Example** :TRIGGER:MODE ALEVEL  
:TRIGGER:MODE? -> :TRIGGER:MODE ALEVEL

### **:TRIGger:POSition**

**Function** Sets the trigger position or queries the current setting.

**Syntax** :TRIGger:POSition {<NRf>}  
:TRIGger:POSition?  
<NRf> = 0 to 100 (%)

**Example** :TRIGGER:POSITION 10  
:TRIGGER:POSITION? -> :TRIGGER:  
POSITION 10

### **:TRIGger:SCOUNT (Single(N) Count)**

**Function** Sets the number of times the trigger is to be activated when the trigger mode is Single(N) or queries the current setting.

**Syntax** :TRIGger:SCOUNT {<NRf>}  
:TRIGger:SCOUNT?  
<NRf> = See the DL9000 User's Manual.

**Example** :TRIGGER:SCOUNT 1  
:TRIGGER:SCOUNT? -> :TRIGGER:SCOUNT 1

**:TRIGger:SOURCE?**

**Function** Queries all settings related to the trigger source.

**Syntax** :TRIGger:SOURCE?

**Example** :TRIGGER:SOURCE? -> :TRIGGER:SOURCE:  
CHANNEL1:COUPLING DC;HPREJECTION OFF;  
HYSTERESIS HIGH;LEVEL 1.000E+00;  
STATE HIGH;WIDTH 1.000E+00;WINDOW 0;;  
TRIGGER:SOURCE:CHANNEL2:COUPLING DC;  
HPREJECTION OFF;HYSTERESIS HIGH;  
LEVEL 1.000E+00;STATE HIGH;  
WIDTH 1.000E+00;WINDOW 0;;:TRIGGER:  
SOURCE:CHANNEL3:COUPLING DC;  
HPREJECTION OFF;HYSTERESIS HIGH;  
LEVEL 1.000E+00;STATE HIGH;  
WIDTH 1.000E+00;WINDOW 0;;:TRIGGER:  
SOURCE:CHANNEL4:COUPLING DC;  
HPREJECTION OFF;HYSTERESIS HIGH;  
LEVEL 1.000E+00;STATE HIGH;  
WIDTH 1.000E+00;WINDOW 0;;:TRIGGER:  
SOURCE:EXTERNAL:LEVEL 0.000E+00;  
PROBE 1;;:TRIGGER:SOURCE:LOGIC AND

**:TRIGger:SOURCE:CHANnel<x>?**

**Function** Queries all settings related to the channel of the trigger source.

**Syntax** :TRIGger:SOURCE:CHANnel<x>?  
<x> = 1 to 4

**Example** :TRIGGER:SOURCE:CHANNEL1? -> :TRIGGER:  
SOURCE:CHANNEL1:COUPLING DC;  
HPREJECTION OFF;HYSTERESIS HIGH;  
LEVEL 1.000E+00;STATE HIGH;  
WIDTH 1.000E+00;WINDOW 0

**:TRIGger:SOURCE:CHANnel<x>:COUPLing**

**Function** Sets the trigger coupling of the channel or queries the current setting.

**Syntax** :TRIGger:SOURCE:CHANnel<x>:  
COUPLing {AC|DC}  
:TRIGger:SOURCE:CHANnel<x>:COUPLing?  
<x> = 1 to 4

**Example** :TRIGGER:SOURCE:CHANNEL1:COUPLING AC  
:TRIGGER:SOURCE:CHANNEL1:COUPLING?  
-> :TRIGGER:SOURCE:CHANNEL1:COUPLING DC

**:TRIGger:SOURCE:CHANnel<x>:****HFRejection (HighFrequencyREJECTION)**

**Function** Sets the low pass filter (HF rejection) of the channel or queries the current setting.

**Syntax** :TRIGger:SOURCE:CHANnel<x>:  
HFRejection {<Frequency>|OFF}  
:TRIGger:SOURCE:CHANnel<x>:HFRejection?  
<x> = 1 to 4  
<Frequency> = 20MHz or 15kHz

**Example** :TRIGGER:SOURCE:CHANNEL1:  
HPREJECTION OFF  
:TRIGGER:SOURCE:CHANNEL1:HFREJECTION?  
-> :TRIGGER:SOURCE:CHANNEL1:  
HPREJECTION OFF

**Description** This command is invalid when the trigger source is {EXTERNAL|LINE}.

**:TRIGger:SOURCE:CHANnel<x>:HYSTeresis**

**Function** Sets the hysteresis of the channel or queries the current setting.

**Syntax** :TRIGger:SOURCE:CHANnel<x>:  
HYSTeresis {HIGH|LOW}  
:TRIGger:SOURCE:CHANnel<x>:HYSTeresis?  
<x> = 1 to 4

**Example** :TRIGGER:SOURCE:CHANNEL1:  
HYSTERESIS HIGH  
:TRIGGER:SOURCE:CHANNEL1:HYSTERESIS?  
-> :TRIGGER:SOURCE:CHANNEL1:  
HYSTERESIS HIGH

**:TRIGger:SOURCE:CHANnel<x>:LEVel**

**Function** Sets the trigger level of the channel or queries the current setting.

**Syntax** :TRIGger:SOURCE:CHANnel<x>:  
LEVel {<Voltage>|<Current>}  
:TRIGger:SOURCE:CHANnel<x>:LEVel?  
<x> = 1 to 4  
<Voltage> and <Current> = See the DL9000 User's Manual.

**Example** :TRIGGER:SOURCE:CHANNEL1:LEVEL 1V  
:TRIGGER:SOURCE:CHANNEL1:LEVEL?  
-> :TRIGGER:SOURCE:CHANNEL1:  
LEVEL 1.000E+00

## 5.29 TRIGger Group

### **:TRIGger:SOURce:CHANnel<x>:STATE**

**Function** Sets the condition to be satisfied of the channel or queries the current setting.

**Syntax** :TRIGger:SOURce:CHANnel<x>:  
STATE {DONTcare|HIGH|IN|LOW|OUT}  
:TRIGger:SOURce:CHANnel<x>:STATE?  
<x> = 1 to 4

**Example** :TRIGGER:SOURCE:CHANNEL1:STATE HIGH  
:TRIGGER:SOURCE:CHANNEL1:STATE?  
-> :TRIGGER:SOURCE:CHANNEL1:STATE HIGH

**Description** • This command is valid when :TRIGger:TYPE EQUalify|I2Cbus|PQUalify|PSTATE|SPATtern|STATE.  
• {HIGH|LOW} is valid when :TRIGger:TYPE I2Cbus|SPATtern.  
• {IN|OUT} is valid when :TRIGger:TYPE EQUalify|PQUalify|PSTATE|STATE and :TRIGger:SOURce:CHANnel<x>:WINDow ON. {HIGH|LOW} is valid when :TRIGger:TYPE EQUalify|PQUalify|PSTATE|STATE and :TRIGger:SOURce:CHANnel<x>:WINDow OFF.

### **:TRIGger:SOURce:CHANnel<x>:WIDTh**

**Function** Sets the window trigger width of the channel or queries the current setting.

**Syntax** :TRIGger:SOURce:CHANnel<x>:  
WIDTh {<Voltage>|<Current>}  
:TRIGger:SOURce:CHANnel<x>:WIDTh?  
<x> = 1 to 4  
<Voltage> and <Current> = See the DL9000 User's Manual.

**Example** :TRIGGER:SOURCE:CHANNEL1:WIDTH 1V  
:TRIGGER:SOURCE:CHANNEL1:WIDTH?  
-> :TRIGGER:SOURCE:CHANNEL1:  
WIDTH 1.000E+00

**Description** This command is valid when :TRIGger:SOURce:CHANnel<x>:WINDow ON.

### **:TRIGger:SOURce:CHANnel<x>:WINDow**

**Function** Turns ON/OFF the window of the channel or queries the current setting.

**Syntax** :TRIGger:SOURce:CHANnel<x>:  
WINDow {<Boolean>}  
:TRIGger:SOURce:CHANnel<x>:WINDow?  
<x> = 1 to 4

**Example** :TRIGGER:SOURCE:CHANNEL1:WINDOW ON  
:TRIGGER:SOURCE:CHANNEL1:WINDOW?  
-> :TRIGGER:SOURCE:CHANNEL1:WINDOW 1

### **:TRIGger:SOURce:EXTErnal?**

**Function** Queries all settings related to the external trigger.

**Syntax** :TRIGger:SOURce:EXTErnal?  
**Example** :TRIGGER:SOURCE:EXTERNAL? -> :TRIGGER:  
SOURCE:EXTERNAL:LEVEL 0.000E+00;PROBE 1

### **:TRIGger:SOURce:EXTErnal:LEVEl**

**Function** Sets the trigger level of the external trigger or queries the current setting.

**Syntax** :TRIGger:SOURce:EXTErnal:  
LEVEl {<Voltage>|<Current>}  
:TRIGger:SOURce:EXTErnal:LEVEl?  
<x> = 1 to 4

**Example** :TRIGGER:SOURCE:EXTERNAL:LEVEL 1V  
:TRIGGER:SOURCE:EXTERNAL:LEVEL?  
-> :TRIGGER:SOURCE:EXTERNAL:  
LEVEL 1.000E+00

**Description** This command is valid when :TRIGger:TYPE EDGE|EQUalify|PQUalify|PULSE.

### **:TRIGger:SOURce:EXTErnal:PROBE**

**Function** Sets the probe attenuation of the external trigger or queries the current setting.

**Syntax** :TRIGger:SOURce:EXTErnal:PROBE {<NRF>}  
:TRIGger:SOURce:EXTErnal:PROBE?  
<NRF> = 1,10

**Example** :TRIGGER:SOURCE:EXTERNAL:PROBE 1  
:TRIGGER:SOURCE:EXTERNAL:PROBE?  
-> :TRIGGER:SOURCE:EXTERNAL:PROBE 1

**Description** This command is valid when :TRIGger:TYPE EDGE|EQUalify|PQUalify|PULSE.

### **:TRIGger:SOURce:LOGic**

**Function** Sets the trigger source logic or queries the current setting.

**Syntax** :TRIGger:SOURce:LOGic {AND|OR}  
:TRIGger:SOURce:LOGic?

**Example** :TRIGGER:SOURCE:LOGIC AND  
:TRIGGER:SOURCE:LOGIC? -> :TRIGGER:  
SOURCE:LOGIC AND

**Description** This command is valid when :TRIGger:TYPE EQUalify|I2Cbus|PQUalify|PSTATE|SPATtern|STATE.

### **:TRIGger:TYPE**

**Function** Sets the trigger type or queries the current setting.

**Syntax** :TRIGger:TYPE {CANbus|EDGE|EICycle|EIDelay|EISequence|EOR|EQUalify|I2Cbus|LEDGe|LINbus|PQUalify|PSTATE|PULSE|SPATtern|SPIbus|STATE|TV|UART}  
:TRIGger:TYPE?

**Example** :TRIGGER:TYPE CANBUS:TRIGGER:TYPE?  
-> :TRIGGER:TYPE CANBUS

### **:TRIGger:WIDTh?**

**Function** Queries all settings related to the pulse width trigger.

**Syntax** :TRIGger:WIDTh?  
**Example** :TRIGGER:WIDTH? -> :TRIGGER:WIDTH:  
MODE OUT;POLARITY POSITIVE;SOURCE 1;  
TIME1 1.000E-09;TIME2 1.000E-09



**: TRIGger: WIDTH: MODE**

**Function** Sets the determination mode of the pulse width trigger or queries the current setting.

**Syntax** :TRIGger:WIDTH:MODE {BETween|IN|NOTBetween|OUT|TIMEout}  
:TRIGger:WIDTH:MODE?

**Example** :TRIGGER:WIDTH:MODE BETWEEN  
:TRIGGER:WIDTH:MODE? -> :TRIGGER:WIDTH:MODE BETWEEN

**: TRIGger: WIDTH: POLarity**

**Function** Sets the polarity of the pulse width trigger or queries the current setting.

**Syntax** :TRIGger:WIDTH:POLarity {FALSe|IN|NEGative|OUT|POSitive|TRUE}  
:TRIGger:WIDTH:POLarity?

**Example** :TRIGGER:WIDTH:POLARITY POSITIVE  
:TRIGGER:WIDTH:POLARITY? -> :TRIGGER:WIDTH:POLARITY POSITIVE

**Description** • {IN|OUT} is valid when :TRIGger:TYPE PQUalify|PULSe and :TRIGger:SOURce:CHANnel<x>:WINDow ON. {HIGH|LOW} is valid when :TRIGger:TYPE PQUalify|PULSe and :TRIGger:SOURce:CHANnel<x>:WINDow OFF.  
• {FALSe|TRUE} is valid when :TRIGger:TYPE PSTATE.

**: TRIGger: WIDTH: SOURce**

**Function** Sets the trigger source of the pulse width trigger or queries the current setting.

**Syntax** :TRIGger:WIDTH:SOURce {<NRf>|EXTernal}  
:TRIGger:WIDTH:SOURce?  
<NRf> = 1 to 4

**Example** :TRIGGER:WIDTH:SOURCE EXTERNAL  
:TRIGGER:WIDTH:SOURCE? -> :TRIGGER:WIDTH:SOURCE EXTERNAL

**Description** This command is valid when :TRIGger:TYPE PQUalify|PULSe.

**: TRIGger: WIDTH: TIME<x>**

**Function** Sets the pulse width of the pulse width trigger or queries the current setting.

**Syntax** :TRIGger:WIDTH:TIME<x> {<Time>}  
:TRIGger:WIDTH:TIME<x>?  
<x> = 1 or 2

<Time> = 1 ns to 10 s in 500-ps steps

**Example** :TRIGGER:WIDTH:TIME1 1S  
:TRIGGER:WIDTH:TIME1? -> :TRIGGER:WIDTH:TIME1 1.000E+00

**Description** TIME2 is valid when :TRIGger:WIDTH:MODE BETWween|NOTBetween.

## 5.30 WAVEform Group

The commands in this group deal with acquired waveform data. There are no front panel keys that correspond to the commands in this group.

### **:WAVEform?**

Function Queries all information about the waveform data.  
Syntax :WAVEform?  
Example :WAVEFORM? -> :WAVEFORM:TRACE 1;  
RECORD 0;START 0;END 6249999;  
FORMAT WORD;  
BYTEORDER LSBFIRST

### **:WAVEform:BITS?**

Function Queries the bit length of the waveform data specified by “:WAVEform:TRACE”.  
Syntax :WAVEform:BITS?  
Example :WAVEFORM:BITS? -> :WAVEFORM:BITS 16

### **:WAVEform:BYTeorder**

Function Sets the transmission order when using word format of two bytes or more or queries the current setting.  
Syntax :WAVEform:BYTeorder {LSBFirst|MSBFirst}  
:WAVEform:BYTeorder?  
Example :WAVEFORM:BYTEORDER LSBFIRST  
:WAVEFORM:BYTEORDER? -> :WAVEFORM:BYTEORDER LSBFIRST

### **:WAVEform:END**

Function Sets the last data point of the waveform specified by :WAVEform:TRACE or queries the current setting.  
Syntax :WAVEform:END {<NRf>}  
:WAVEform:END?  
<NRf> = 0 to 6,249,999  
Example :WAVEFORM:END 12499  
:WAVEFORM:END? -> :WAVEFORM:END 12499  
Description The total number of data points can be queried using :WAVEform:LENGth?.

### **:WAVEform:FORMat**

Function Sets the format of the data to be transmitted or queries the current setting.  
Syntax :WAVEform:FORMat {ASCIi|BYTE|RBYTe|WORD}  
:WAVEform:FORMat?  
Example :WAVEFORM:FORMAT ASCII  
:WAVEFORM:FORMAT? -> :WAVEFORM:FORMAT ASCII  
Description For details on the differences in the format setting, see the description of :WAVEform:SEND?.

### **:WAVEform:LENGth?**

Function Queries the total number of points of the waveform specified by “:WAVEform:TRACE”.  
Syntax :WAVEform:LENGth?  
Example :WAVEFORM:LENGTH? -> :WAVEFORM:LENGTH 12500

### **:WAVEform:OFFSet?**

Function Queries the offset value when converting the waveform data specified by :WAVEform:TRACE to physical values.

Syntax :WAVEform:OFFSet?  
Example :WAVEFORM:OFFSET? -> 0.000E+00

Description • The offset value is used when converting the <Block data> that is output using :WAVEform:SEND? to physical values.

- When :CHANnel<x>:OCANcel is ON, 0 is returned.

### **:WAVEform:POSition?**

Function Queries the vertical axis position used for converting to voltage when RBYTe is specified with :WAVEform:FORMat.

Syntax :WAVEform:POSition?  
Example :WAVEFORM:POSITION? -> :WAVEFORM:POSITION 128

### **:WAVEform:RANGe?**

Function Queries the range value when converting the waveform data specified by :WAVEform:TRACE to physical values.

Syntax :WAVEform:RANGe?  
Example :WAVEform:RANGe? -> 5.000E+00

Description The range value is used when converting the <Block data> that is output using :WAVEform:SEND? to physical values.

### **:WAVEform:RECOrd**

Function Sets the target record number for the commands in the WAVEform group or queries the current setting.

Syntax :WAVEform:RECOrd  
{AVERAge|MINimum|<NRf>}  
:WAVEform:RECOrd?  
<NRf> = 0 to -1999

Example :WAVEFORM:RECORD 0  
:WAVEFORM:RECORD? -> :WAVEFORM:RECORD 0

Description • If “AVERAge” is specified, the commands in the WAVEform group are applied to the average value of the history waveform. The record numbers to be averaged are set using the “:HISTory[:CURRent]:DISPlay” command. In addition, the highlight display mode must be set to “AVERAge.” Set the highlight display mode using the “:HISTory[:CURRent]:MODE” command.

- Specifying “MINimum” sets the record to the minimum record number. The selectable record number varies depending on the model and acquisition setting. For details, see the DL9000 User’s Manual.

**:WAVEform:RECORD? MINimum**

Function Queries the minimum record number of the history of the target channel.

Syntax :WAVEform:RECORD? MINimum

Example :WAVEFORM:RECORD? MINimum -> :WAVEFORM:RECORD -1999

**:WAVEform:SEND?**

Function Queries the waveform data specified by “:WAVEform:TRACE”.

Syntax :WAVEform:SEND? [{<NRf>}]  
<NRf> = 1 to 2000

Varies depending on the record length setting.

Example :WAVEFORM:SEND? -> #8 (number of bytes, 8 digits) (data sequence)  
or <NRf>,<NRf>,...

Description • The output format of :WAVEform:SEND? varies depending on the :WAVEform:FORMat setting.

(1) When set to ASCII

Returned in the following format: <Voltage>,<Voltage>, ... <Voltage>.

(2) When set to BYTE or WORD

Returned in the <Block data> format.

You can convert the value using the following equation.

Voltage (computed value) = (range × data / divisions\*) + offset

\* BYTE: Divisions = 12.5

WORD: Divisions = 3200

(3) When set to RBYTE

Returned in the <Block data> format.

You can convert the value using the following equation.

Voltage (computed value) = (range × (data - Position) / divisions\*) + offset

Divisions = 12.5

Position = Return value of “:WAVEform:POSition?”.

- <NRf> can be omitted. If <NRf> is attached, waveform data is queried <NRf> times in order from the record number specified by :WAVEform:RECORD - <NRf> + 1.

**:WAVEform:SIGN?**

Function Queries the existence of a sign when querying the waveform data specified by :WAVEform:TRACE using binary data.

Syntax :WAVEform:SIGN?

Example :WAVEFORM:SIGN? -> :WAVEFORM:SIGN 1

**:WAVEform:SRATE? (Sample RATE)**

Function Queries the sample rate of the record specified by :WAVEform:RECORD.

Syntax :WAVEform:SRATE?

Example :WAVEFORM:SRATE? -> :WAVEFORM:SRATE 1.25E+09

**:WAVEform:START**

Function Sets the first data point of the waveform specified by :WAVEform:TRACE or queries the current setting.

Syntax :WAVEform:START {<NRf>}

:WAVEform:START?

<NRf> = 0 to 6,249,999 (0 to 2,499,999 on 2.5 MW memory models)

Example :WAVEFORM:START 0

:WAVEFORM:START? -> :WAVEFORM:START 0

**:WAVEform:TRACE**

Function Sets the target waveform or queries the current setting.

Syntax :WAVEform:TRACE {<NRf>|MATH<x>|

REFERENCE<x>}

:WAVEform:TRACE?

<NRf> = 1 to 4

<x> of MATH<x> = 1 to 8

<x> of REFERENCE<x> = 1 to 4

Example :WAVEFORM:TRACE 1

:WAVEFORM:TRACE? -> :WAVEFORM:TRACE 1

**:WAVEform:TRIGGER?**

Function Queries the trigger position of the record specified by :WAVEform:RECORD.

Syntax :WAVEform:TRIGGER?

Example :WAVEFORM:TRIGGER? -> :WAVEFORM:TRIGGER 6250

Description Queries the number of points from the first point of the record length to the trigger position.

**:WAVEform:TYPE?**

Function Queries the acquisition mode of the waveform specified by :WAVEform:TRACE.

Syntax :WAVEform:TYPE?

Example :WAVEFORM:TYPE? -> :WAVEFORM:TYPE NORMAL

## 5.31 ZOOM Group

### : ZOOM?

**Function** Queries all settings related to the waveform zoom.

**Syntax** :ZOOM?

**Example** :ZOOM? -> :ZOOM:ALLOCATION1:TRACE1 1;  
TRACE2 1;TRACE3 1;TRACE4 1;TRACE5 1;  
TRACE6 1;TRACE7 1;TRACE8 1;:ZOOM:  
ALLOCATION2:TRACE1 1;TRACE2 1;TRACE3 1;  
TRACE4 1;TRACE5 1;TRACE6 1;TRACE7 1;  
TRACE8 1;:ZOOM:FORMAT1 MAIN;  
FORMAT2 MAIN;HLINKAGE 0;HORIZONTAL1:  
ASCROLL:SPEED 5;:ZOOM:HORIZONTAL1:  
MAG 2.000E+00;POSITION 0.000E+00;:ZOOM:  
HORIZONTAL2:ASCROLL:SPEED 5;:ZOOM:  
HORIZONTAL2:MAG 2.000E+00;  
POSITION 0.000E+00;:ZOOM:MODE MAIN;  
TYPE1 HORIZONTAL;TYPE2 HORIZONTAL;  
VERTICAL1:MAG 1.000E+00;  
POSITION 0.000E+00;TRACE 1;:ZOOM:  
VERTICAL2:MAG 1.000E+00;  
POSITION 0.000E+00;TRACE 1;:ZOOM:  
VLINKAGE 0

### : ZOOM:ALLOCATION<x>?

**Function** Queries all settings related to the zoom source waveform.

**Syntax** :ZOOM:ALLOCATION<x>?  
<x> = 1 or 2

**Example** :ZOOM:ALLOCATION1? -> :ZOOM:ALLOCATION1:  
TRACE1 1;TRACE2 1;  
TRACE3 1;TRACE4 1;TRACE5 1;TRACE6 1;  
TRACE7 1;TRACE8 1

### : ZOOM:ALLOCATION<x>:ALLON

**Function** Sets all waveforms to be zoomed.

**Syntax** :ZOOM:ALLOCATION<x>:ALLON  
<x> = 1 or 2

**Example** :ZOOM:ALLOCATION1:ALLON

### : ZOOM:ALLOCATION<x>:TRACE<x>

**Function** Turns ON/OFF the trace you wish to zoom or queries the current setting.

**Syntax** :ZOOM:ALLOCATION<x>:  
TRACE<x> {<Boolean>}  
:ZOOM:ALLOCATION<x>:TRACE<x>?  
<x> of ALLOCATION<x> = 1 or 2  
<x> of TRACE<x> = 1 to 8

**Example** :ZOOM:ALLOCATION1:TRACE1 ON  
:ZOOM:ALLOCATION1:TRACE1? -> :ZOOM:  
ALLOCATION1:TRACE1 1

### : ZOOM:FORMAt<x>

**Function** Sets the display format of the zoom waveform or queries the current setting.

**Syntax** :ZOOM:FORMAt<x> {DUAL|MAIN|QUAD|SINGLE|  
TRIad}  
:ZOOM:FORMAt<x>?  
<x> = 1 or 2

**Example** :ZOOM:FORMAT1 SINGLE  
:ZOOM:FORMAT1? -> :ZOOM:FORMAT1 SINGLE

### : ZOOM:HLINKage

**Function** Turns ON/OFF the horizontal link or queries the current setting.

**Syntax** :ZOOM:HLINKage {<Boolean>}  
:ZOOM:HLINKage?

**Example** :ZOOM:HLINKAGE ON  
:ZOOM:HLINKAGE? -> :ZOOM:HLINKAGE 1

### : ZOOM:HORizontal<x>?

**Function** Queries all settings related to the horizontal zoom.

**Syntax** :ZOOM:HORizontal<x>?  
<x> = 1 or 2

**Example** :ZOOM:HORIZONTAL1? -> :ZOOM:HORIZONTAL1:  
ASCROLL:SPEED 5;:  
ZOOM:HORIZONTAL1:MAG 2.000E+00;  
POSITION 4.000E+00

### : ZOOM:HORizontal<x>:ASCROLL?

**Function** Queries all settings related to the auto scroll function.

**Syntax** :ZOOM:HORizontal<x>:ASCROLL?  
<x> = 1 or 2

**Example** :ZOOM:HORIZONTAL1:ASCROLL? -> :ZOOM:  
HORIZONTAL1:ASCROLL:SPEED 5

### : ZOOM:HORizontal<x>:ASCROLL:JUMP

**Function** Moves the zoom center position to the left or right edge of the main screen.

**Syntax** :ZOOM:HORizontal<x>:ASCROLL:JUMP {LEFT|  
RIGHT}  
<x> = 1 or 2

**Example** :ZOOM:HORIZONTAL1:ASCROLL:JUMP RIGHT

### : ZOOM:HORizontal<x>:ASCROLL:SPEEd

**Function** Sets the auto scroll speed or queries the current setting.

**Syntax** :ZOOM:HORizontal<x>:ASCROLL:SPEEd  
{<Nrf>}  
:ZOOM:HORizontal<x>:ASCROLL:SPEEd?  
<x> = 1 or 2  
<Nrf> = 1, 2, 5, 10, 20, or 50

**Example** :ZOOM:HORIZONTAL1:ASCROLL:SPEED 1  
:ZOOM:HORIZONTAL1:ASCROLL:SPEED?  
-> :ZOOM:HORIZONTAL1:ASCROLL:SPEED 1

**: ZOOM: HOrizontal<x>: ASCRoll: START**

Function Starts auto scrolling.

Syntax : ZOOM: HOrizontal<x>: ASCRoll: START  
{ LEFT|RIGHT}  
<x> = 1 or 2

Example : ZOOM: HORIZONTAL1: ASCROLL: START LEFT

**: ZOOM: HOrizontal<x>: ASCRoll: STOP**

Function Stops auto scrolling.

Syntax : ZOOM: HOrizontal<x>: ASCRoll: STOP  
<x> = 1 or 2

Example : ZOOM: HORIZONTAL1: ASCROLL: STOP

**: ZOOM: HOrizontal<x>: MAG**

Function Sets the horizontal zoom magnification or queries the current setting.

Syntax : ZOOM: HOrizontal<x>: MAG {<NRF>}  
: ZOOM: HOrizontal<x>: MAG?  
<x> = 1 or 2

<NRF> = See the DL9000 User's Manual.

Example : ZOOM: HORIZONTAL1: MAG 2  
: ZOOM: HORIZONTAL1: MAG? -> : ZOOM:  
HORIZONTAL1: MAG 2.000E+00

**: ZOOM: HOrizontal<x>: POSition**

Function Sets the horizontal zoom center position or queries the current setting.

Syntax : ZOOM: HOrizontal<x>: POSition {<NRF>}  
: ZOOM: HOrizontal<x>: POSition?  
<x> = 1 or 2  
<NRF> = -5 to 5 (div)

Example : ZOOM: HORIZONTAL1: POSITION 1  
: ZOOM: HORIZONTAL1: POSITION? -> : ZOOM:  
HORIZONTAL1: POSITION 1.000E+00

**: ZOOM: MODE**

Function Sets the zoom waveform display format or queries the current setting.

Syntax : ZOOM: MODE {MAIN|MAIN\_Z1|MAIN\_Z1\_Z2|  
MAIN\_Z2|Z1|Z1\_Z2|Z2}  
: ZOOM: MODE?

Example : ZOOM: MODE MAIN\_Z1\_Z2  
: ZOOM: MODE? -> : ZOOM: MODE MAIN\_Z1\_Z2

**: ZOOM: TYPE<x>**

Function Sets the zoom type or queries the current setting.

Syntax : ZOOM: TYPE<x> {HOrizontal|VERTical}  
: ZOOM: TYPE<x>?  
<x> = 1 or 2

Example : ZOOM: TYPE1 VERTICAL  
: ZOOM: TYPE1? -> : ZOOM: TYPE1 VERTICAL

**: ZOOM: VERTical<x>?**

Function Queries all settings related to the vertical zoom.

Syntax : ZOOM: VERTical<x>?  
<x> = 1 or 2

Example : ZOOM: VERTICAL1? -> : ZOOM: VERTICAL1:  
MAG 1.000E+00; POSITION 0.000E+00;  
TRACE 1

**: ZOOM: VERTical<x>: INITIalize**

Function Initializes the vertical zoom.

Syntax : ZOOM: VERTical<x>: INITIalize  
<x> = 1 or 2

Example : ZOOM: VERTICAL1: INITIALIZE

**: ZOOM: VERTical<x>: MAG**

Function Sets the vertical zoom magnification or queries the current setting.

Syntax : ZOOM: VERTical<x>: MAG {<NRF>}  
: ZOOM: VERTical<x>: MAG?  
<x> = 1 or 2

<NRF> = See the DL9000 User's Manual.

Example : ZOOM: VERTICAL1: MAG 1  
: ZOOM: VERTICAL1: MAG? -> : ZOOM: VERTICAL1:  
MAG 1.000E+00

**: ZOOM: VERTical<x>: POSition**

Function Sets the vertical zoom position or queries the current setting.

Syntax : ZOOM: VERTical<x>: POSition {<NRF>}  
: ZOOM: VERTical<x>: POSition?  
<x> = 1 or 2  
<NRF> = -4 to 4 (div)

Example : ZOOM: VERTICAL1: POSITION 1  
: ZOOM: VERTICAL1: POSITION? -> : ZOOM:  
VERTICAL1: POSITION 1.000E+00

**: ZOOM: VERTical<x>: TRAcE**

Function Sets the trace you wish to display on the vertical zoom screen or queries the current setting.

Syntax : ZOOM: VERTical<x>: TRAcE {<NRF>}  
: ZOOM: VERTical<x>: TRAcE?  
<x> = 1 or 2  
<NRF> = 1 to 8

Example : ZOOM: VERTICAL1: TRACE 1  
: ZOOM: VERTICAL1: TRACE? -> : ZOOM:  
VERTICAL1: TRACE 1

**: ZOOM: VLINKage**

Function Turns ON/OFF the vertical link or queries the current setting.

Syntax : ZOOM: VLINKage {<Boolean>}  
: ZOOM: VLINKage?

Example : ZOOM: VLINKAGE ON  
: ZOOM: VLINKAGE? -> : ZOOM: VLINKAGE 1

## 5.32 Common Command Group

The commands in the common group are defined in the USBTMC-USB488 and are independent of the instrument's functions. There are no front panel keys that correspond to the commands in this group.

### **\*CAL? (CALibrate)**

**Function** Performs calibration and queries the result.  
**Syntax** \*CAL?  
**Example** \*CAL? -> 0  
**Description** If the calibration terminates normally, 0 is returned. If an error is detected, 1 is returned.

### **\*CLS (CLear Status)**

**Function** Clears the standard event register, extended event register, and error queue.  
**Syntax** \*CLS  
**Example** \*CLS  
**Description** • If the \*CLS command is located immediately after the program message terminator, the output queue is also cleared.  
• For details on the register and queue, see chapter 6.

### **\*ESE (standard Event Status Enable register)**

**Function** Sets the standard event enable register or queries the current setting.  
**Syntax** \*ESE {<NRf>}  
\*ESE?  
<NRf> = 0 to 255  
**Example** \*ESE 251  
\*ESE? -> 251  
**Description** • Specify the value as a sum of decimal values of each bit.  
• For example, specifying “\*ESE 251” will cause the standard enable register to be set to “11111011.” In this case, bit 2 of the standard event register is disabled which means that bit 5 (ESB) of the status byte register is not set to 1, even if a “query error” occurs.  
• The default value is “\*ESE 0” (all bits disabled).  
• A query using \*ESE? will not clear the contents of the standard event enable register.  
• For details on the standard event enable register, see page 6-4.

### **\*ESR? (standard Event Status Register)**

**Function** Queries the standard event register and clears the register.  
**Syntax** \*ESR?  
**Example** \*ESR? -> 32  
**Description** • A sum of decimal values of each bit is returned.  
• You can check what type of events occurred when an SRQ is generated.  
• For example, if a value of “32” is returned, this indicates that the standard event register is set to “00100000.” In this case, you can see that the SRQ occurred due to a “command syntax error.”  
• A query using \*ESR? will clear the contents of the standard event register.  
• For details on the standard event register, see page 6-4.

### **\*IDN? (IDeNtify)**

**Function** Queries the instrument model.  
**Syntax** \*IDN?  
**Example** \*IDN? -> YOKOGAWA,701313,27E100000,  
F1.10  
**Description** The information is returned in the following form: <Manufacturer>,<Model>,<Serial No.>,<Firmware version> The values 701307, 701308, 701310, 701311, 701312, and 701313 are returned for the <Model> when the instrument is the DL9040, DL9040L, DL9140, DL9140L, DL9240, and DL9240L, respectively.

**\*LRN? (LeaRN)**

**Function** Queries collectively the current settings of the following command groups.

ACQuire, CHANnel<x>, TIMEbase, TRIGger

**Syntax** \*LRN?

**Example** \*LRN? -> :ACQUIRE:AVERAGE:COUNT 2;  
 EWEIGHT 16;:ACQUIRE:HRMODE 0;  
 INTERLEAVE 0;INTERPOLATE 1;MODE NORMAL;  
 REPETITIVE 0;RLENGTH 12500;:CHANNEL1:  
 SELECT INPUT;DISPLAY 1;BWIDTH FULL;  
 COUPLING DC;DESKEW 0.000E+00;INVERT 0;  
 LABEL:DEFINE "CH1";MODE 1;:CHANNEL1:  
 OCANCEL 0;OFFSET 0.000E+00;  
 POSITION 0.000E+00;PROBE:MODE 1;:  
 CHANNEL1:SVALUE 0;VDIV 1.000E+00;:  
 CHANNEL2:SELECT INPUT;DISPLAY 1;  
 BWIDTH FULL;COUPLING DC;  
 DESKEW 0.000E+00;INVERT 0;LABEL:  
 DEFINE "CH2";MODE 1;:CHANNEL2:  
 OCANCEL 0;OFFSET 0.000E+00;  
 POSITION 0.000E+00;PROBE:MODE 1;:  
 CHANNEL2:SVALUE 0;VDIV 1.000E+00;:  
 CHANNEL3:SELECT INPUT;DISPLAY 1;  
 BWIDTH FULL;COUPLING DC;  
 DESKEW 0.000E+00;INVERT 0;LABEL:  
 DEFINE "CH3";MODE 1;:CHANNEL3:  
 OCANCEL 0;OFFSET 0.000E+00;  
 POSITION 0.000E+00;PROBE:MODE 1;:  
 CHANNEL3:SVALUE 0;VDIV 1.000E+00;:  
 CHANNEL4:SELECT INPUT;DISPLAY 1;  
 BWIDTH FULL;COUPLING DC;  
 DESKEW 0.000E+00;INVERT 0;LABEL:  
 DEFINE "CH4";MODE 1;:CHANNEL4:  
 OCANCEL 0;OFFSET 0.000E+00;  
 POSITION 0.000E+00;PROBE:MODE 1;:  
 CHANNEL4:SVALUE 0;VDIV 1.000E+00;:  
 TIMEBASE:TDIV 1.000E-06;:TRIGGER:  
 ACTION:ACQCOUNT 1;BUZZER 0;HCOPY 0;  
 MODE OFF;SAVE 0;:TRIGGER:TYPE EDGE;  
 CLOCK:SOURCE 1;POLARITY RISE;:TRIGGER:  
 DELAY:EDGE:COUNT 1;:TRIGGER:DELAY:  
 MODE 0;POLARITY RISE;SOURCE 1;  
 TIME 0.000E+00;TYPE BYTIME;:TRIGGER:  
 EINTERVAL:EVENT1:TYPE EDGE;CLOCK:  
 SOURCE 1;POLARITY RISE;:TRIGGER:  
 EINTERVAL:EVENT1:ESTATE:SOURCE 1;  
 POLARITY RISE;:TRIGGER:EINTERVAL:  
 EVENT1:STATE:CHANNEL1 DONTCARE;  
 CHANNEL2 DONTCARE;CHANNEL3 DONTCARE;  
 CHANNEL4 DONTCARE;LOGIC AND;:TRIGGER:  
 EINTERVAL:EVENT1:WIDTH:MODE OUT;  
 POLARITY POSITIVE;SOURCE 1;  
 TIME1 1.000E-09;TIME2 2.000E-09;:  
 TRIGGER:EINTERVAL:EVENT2:TYPE EDGE;  
 CLOCK:SOURCE 1;POLARITY RISE;:TRIGGER:  
 EINTERVAL:EVENT2:ESTATE:SOURCE 1;

POLARITY RISE;:TRIGGER:EINTERVAL:  
 EVENT2:STATE:CHANNEL1 DONTCARE;  
 CHANNEL2 DONTCARE;CHANNEL3 DONTCARE;  
 CHANNEL4 DONTCARE;LOGIC AND;:TRIGGER:  
 EINTERVAL:EVENT2:WIDTH:MODE OUT;  
 POLARITY POSITIVE;SOURCE 1;  
 TIME1 1.000E-09;TIME2 2.000E-09;:  
 TRIGGER:EINTERVAL:MODE OUT;  
 TIME1 1.500E-09;TIME2 2.000E-09;TRY:  
 MODE 0;SELECT 1;:TRIGGER:ENHANCED:TV:  
 CUSTOMIZE 0;FIELD 1;FRAME 1;HDTV:  
 LINE 2;POLARITY POSITIVE;:TRIGGER:  
 ENHANCED:TV:LEVEL 500.0E-03;NTSC:  
 LINE 5;POLARITY NEGATIVE;:TRIGGER:  
 ENHANCED:TV:PAL:LINE 2;  
 POLARITY NEGATIVE;:TRIGGER:ENHANCED:TV:  
 SGUARD 75;SOURCE 1;TYPE NTSC;  
 USERDEFINE:DEFINITION HD;  
 HFREJECTION OFF;HSYNC 31.500E+03;  
 LINE 2;POLARITY POSITIVE;:TRIGGER:  
 ESTATE:EOR:CHANNEL1 RISE;CHANNEL2 RISE;  
 CHANNEL3 RISE;CHANNEL4 RISE;:TRIGGER:  
 ESTATE:SOURCE 1;POLARITY RISE;:TRIGGER:  
 HOLDOFF 20.00E-09;MODE AUTO;  
 POSITION 50;SCOUNT 1;SOURCE:CHANNEL1:  
 COUPLING DC;HFREJECTION OFF;  
 HYSTERESIS LOW;LEVEL 0.000E+00;  
 STATE DONTCARE;WIDTH 1.000E+00;  
 WINDOW 0;:TRIGGER:SOURCE:CHANNEL2:  
 COUPLING DC;HFREJECTION OFF;  
 HYSTERESIS LOW;LEVEL 0.000E+00;  
 STATE DONTCARE;WIDTH 1.000E+00;  
 WINDOW 0;:TRIGGER:SOURCE:CHANNEL3:  
 COUPLING DC;HFREJECTION OFF;  
 HYSTERESIS LOW;LEVEL 0.000E+00;  
 STATE DONTCARE;WIDTH 1.000E+00;  
 WINDOW 0;:TRIGGER:SOURCE:CHANNEL4:  
 COUPLING DC;HFREJECTION OFF;  
 HYSTERESIS LOW;LEVEL 0.000E+00;  
 STATE DONTCARE;WIDTH 1.000E+00;  
 WINDOW 0;:TRIGGER:SOURCE:EXTERNAL:  
 LEVEL 0.000E+00;PROBE 1;:TRIGGER:  
 SOURCE:LOGIC AND;:TRIGGER:WIDTH:  
 MODE OUT;POLARITY POSITIVE;SOURCE 1;  
 TIME1 1.000E-09;TIME2 2.000E-09

## 5.32 Common Command Group

### \*OPC (Operation Complete)

**Function** Sets bit 0 (OPC bit) of the standard event register to 1 upon the completion of the specified overlap command.

**Syntax** \*OPC

**Example** \*OPC

**Description**

- For the description regarding how to synchronize the program using \*OPC, see page 4-7.
- The `COMMunicate:OPSE` command is used to specify the overlap command.
- If \*OPC is not the last command of the message, the operation is not guaranteed.

### \*OPC? (Operation Complete)

**Function** If \*OPC? is transmitted and the specified overlap command is completed, ASCII code 1 is returned.

**Syntax** \*OPC?

**Example** \*OPC? -> 1

**Description**

- For the description regarding how to synchronize the program using \*OPC, see page 4-8.
- The `COMMunicate:OPSE` command is used to specify the overlap command.
- If \*OPC? is not the last command of the message, the operation is not guaranteed.

### \*OPT? (OPTION)

**Function** Queries the installed options.

**Syntax** \*OPT?

**Example** \*OPT? -> CH6.25MW, PRINTER, ETHER, HDD, USERDEFINE, I2C, CAN, LIN, SPI, UART, PROBEPOWER, PANALYZE, LXIHDDETHER, LXIETHER

**Description**

- Returns the memory model as well as the presence/absence of the built-in printer, Ethernet, internal hard disk, user-defined computation, I<sup>2</sup>C analysis function, CAN analysis function, LIN analysis function, SPI analysis function, UART analysis function, power supply analysis function, and rear panel probe power, internal hard disk + LXI Ethernet, and LXI Ethernet.
- The “\*OPT?” query must be the last query of the program message. An error occurs if there is a query after this query.

### \*PSC (Power-on Status Clear)

**Function** Sets whether or not to clear the registers below at power on or queries the current setting. The register is cleared when the value rounded to an integer is a non-zero value.

- Standard event enable register
- Extended event enable register
- Transition filter

**Syntax** \*PSC {<NRf>}

\*PSC?

<NRf> = 0 (not clear), non-zero (clear)

**Example** \*PSC 1

\*PSC? -> 1

**Description** For details on the registers, see chapter 6.

### \*RST (ReSeT)

**Function** Initializes the settings.

**Syntax** \*RST

**Example** \*RST

**Description** Also clears \*OPC and \*OPC? commands that have been sent earlier.

### \*SRE (Service Request Enable register)

**Function** Sets the service request enable register or queries the current setting.

**Syntax** \*SRE <NRf>

\*SRE?

<NRf> = 0 to 255

**Example** \*SRE 239

\*SRE? -> 239

**Description**

- Specify the value as a sum of decimal values of each bit.
- For example, specifying “\*SRE 239” will cause the service request enable register to be set to “11101111.” In this case, bit 4 of the service request enable register is disabled which means that bit 4 (MAV) of the status byte register is not set to 1, even if “the output queue is not empty.”
- Bit 6 (MSS) of the status byte register is the MSS bit itself, and therefore, is ignored.
- The default value is “\*SRE 0” (all bits disabled).
- A query using \*SRE? will not clear the contents of the service request enable register.
- For details on the service request enable register, see page 6-2.

### \*STB? (STatus Byte)

**Function** Queries the status byte register.

**Syntax** \*STB?

**Example** \*STB? -> 4

**Description**

- The sum of the bits is returned as a decimal value.
- Since the register is read without executing serial polling, bit 6 is a MSS bit not RQS.
- For example, if a value of 4 is returned, this indicates that the status byte register is set to “0000100.” In this case, you can see that “the error queue is not empty” (an error occurred).
- A query using \*STB? will not clear the contents of the status byte register.
- For details on the status byte register, see page 6-3.



**\*TST?**

Function Performs a self-test and queries the result. The self test involves internal memory tests.

Syntax \*TST?

Example \*TST? -> 0

Description If the self-test is successful, 0 is returned. If there is an error, 1 is returned.

**\*WAI (WAIT)**

Function Holds the subsequent command until the completion of the specified overlap operation.

Syntax \*WAI

Example \*WAI

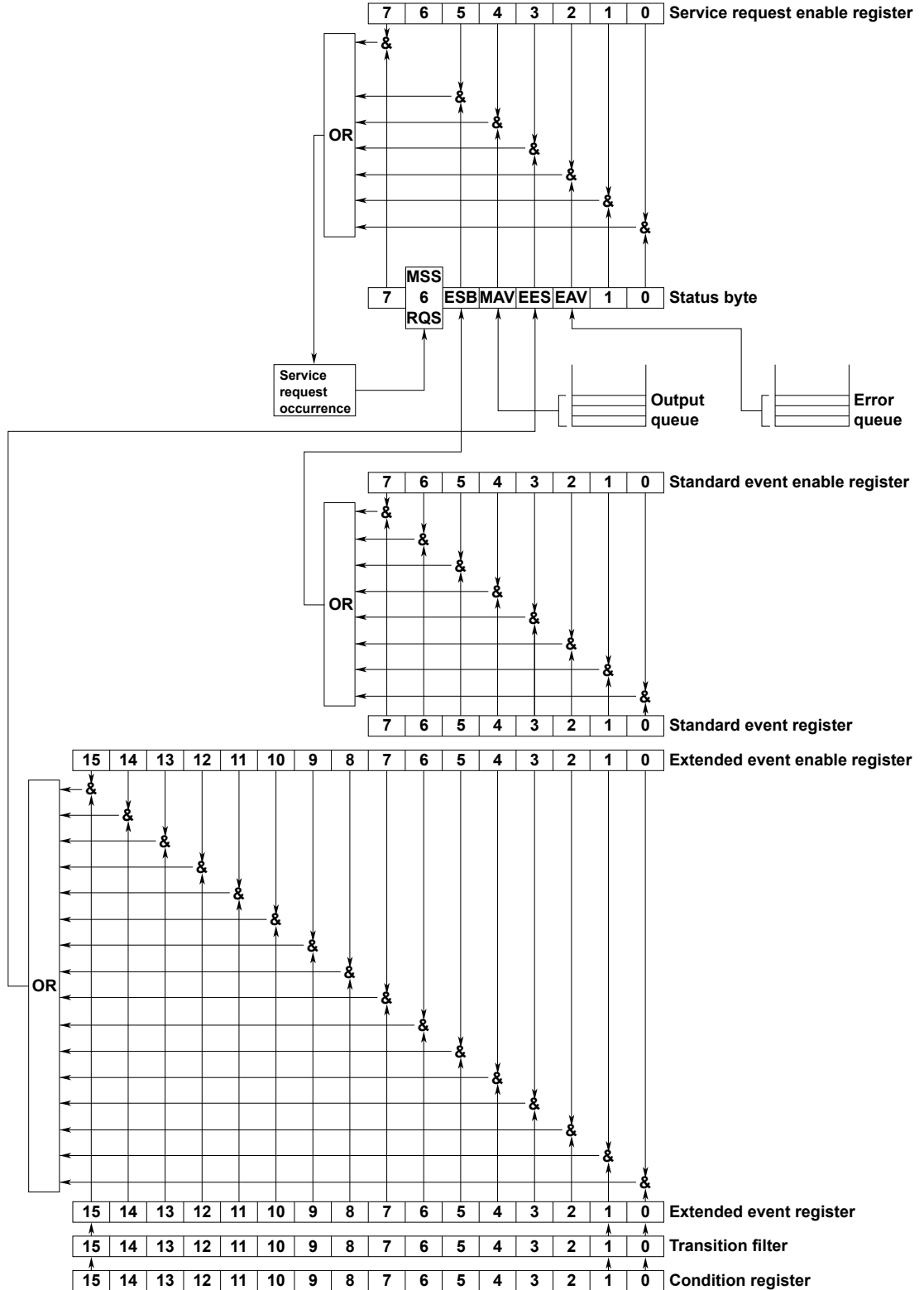
Description • For the description regarding how to synchronize the program using \*WAI, see page 4-7.

- The :COMMunicate:OPSE command is used to specify the overlap command.

# 6.1 Overview of the Status Report

## Status Reports

The figure below shows the status report that is read by serial polling. This status report is an extended version of the status report defined in IEEE 488.2-1992



## 6.1 Overview of the Status Report

### Overview of the Registers and Queues

Name	Functions	Writing	Reading
Status byte	–	–	Serial polling (RQS) *STB? (MSS)
Service request enable register	Masks status byte	*SRE	*SRE?
Standard event register	Changes in device status	–	*ESR?
Standard event enable register	Masks standard event register	*ESE	*ESE?
Extended event register	Changes in device status	–	STATus:EESR?
Extended event enable register	Masks extended event register	STATus:EESE	STATus:EESE?
Condition register	Current instrument status	–	STATus:CONDition?
Transition filter	Conditions that change the extended event register	STATus:FILTer<x>	STATus:FILTer<x>?
Output queue	Stores a response message to a query	All query commands	
Error queue	Stores the error number and message	–	STATus:ERRor?

### Registers and Queues That Affect the Status Byte

Registers that affect the bits of the status byte are shown below.

Standard event register	Sets bit 5 (ESB) of the status byte to 1 or 0.
Output queue	Sets bit 4 (MAV) of the status byte to 1 or 0.
Extended event register	Sets bit 3 (EES) of the status byte to 1 or 0.
Error queue	Sets bit 2 (EAV) of the status byte to 1 or 0.

### Enable Registers

Registers that are used to mask a bit so that the bit will not affect the status byte even when it is set to 1, are shown below.

Status byte	Mask the bits using the service request enable register.
Standard event register	Mask the bits using the standard event enable register.
Extended event register	Mask the bits using the extended event enable register.

### Writing/Reading from Registers

The \*ESE command is used to set the bits in the standard event register to 1's or 0's.

The \*ESE? command is used to query whether the bits in the standard event register are 1's or 0's. For details regarding these commands, see chapter 5.

## 6.2 Status Byte

### Status Byte



- **Bits 0, 1, and 7**  
Not used (always 0)
- **Bit 2 EAV (Error Available)**  
Set to 1 when the error queue is not empty. In other words, this bit is set to 1 when an error occurs. See the page 6-6.
- **Bit 3 EES (Extend Event Summary Bit)**  
Set to 0 when the logical product of the extended event register and the corresponding enable register is 1. In other words, this bit is set to 1 when an event takes place inside the instrument. See the page 6-5.
- **Bit 4 MAV (Message Available)**  
Set to "1" when the output queue is not empty. In other words, this bit is set to 1 when there are data to be transmitted. See the page 6-6.
- **Bit 5 ESB (Event Summary Bit)**  
Set to 0 when the logical product of the standard event register and the corresponding enable register is 1. In other words, this bit is set to 1 when an event takes place inside the instrument. See the page 6-4.
- **Bit 6 RQS(Request Service)/  
MSS(Master Status Summary)**  
Set to 1 when the logical AND of the status byte excluding Bit 6 and the service request enable register is not 0. In other words, this bit is set to 1 when the instrument is requesting service from the controller.  
RQS is set to 1 when the MSS bit changes from 0 to 1, and cleared when serial polling is carried out or when the MSS bit changes to 0.

### Bit Masking

To mask a bit in the status byte so that it does not cause an SRQ, set the corresponding bit of the service request enable register to 0.

For example, to mask bit 2 (EAV) so that service is not requested when an error occurs, set bit 2 of the service request enable register to 0. This can be done using the \*SRE command. To query whether each bit of the service request enable register is 1 or 0, use \*SRE?. For details on the \*SRE command, see chapter 5.

### Operation of the Status Byte

A service request is issued when bit 6 of the status byte becomes 1. Bit 6 is set to 1 when any of the other bits becomes a 1 (when the corresponding bit of the service request enable register is also set to 1).

For example, if an event occurs and the logical AND of the standard event register and the corresponding enable register becomes a 1, then bit 5 (ESB) is set to 1. In this case, if bit 5 of the service request enable register is 1, bit 6 (MSS) will be set to 1, thus requesting service from the controller.

In addition, you can also check what type of event occurred by reading the contents of the status byte.

### Reading from the Status Byte

The following two methods are provided for reading the status byte.

- **Inquiry using the \*STB? query**  
Making an inquiry using the \*STB? query sets bit 6 to MSS. This causes the MSS to be read. After completion of the read-out, none of the bits in the status byte will be cleared.
- **Serial polling**  
Execution of a serial polling changes bit 6 to RQS. This causes RQS to be read. After completion of the read-out, only RQS is cleared. It is not possible to read MSS using serial polling.

### Clearing the Status Byte

No method is provided for forcibly clearing all the bits in the status byte. The bits that are cleared for each operation are shown below.

- **When a query is made using the \*STB? command**  
No bits are cleared.
- **When serial polling is executed**  
Only the RQS bit is cleared.
- **When a \*CLS command is received.**  
When the \*CLS command is received, the status byte itself is not cleared, but the contents of the standard event register (which affects the bits in the status byte) are cleared. As a result, the corresponding bits in the status byte are cleared, except bit 4 (MAV), since the output queue cannot be emptied by the \*CLS command. However, the output queue will also be cleared if the \*CLS command is received just after a program message terminator.

## 6.3 Standard Event Register

### Standard Event Register

7	6	5	4	3	2	1	0
PON	URQ	CME	EXE	DDE	QYE	RQC	OPC

- **Bit 7 PON (Power ON)**  
Set to 1 when the power is turned ON.
- **Bit 6 URQ (User Request)**  
Not used (always 0)
- **Bit 5 CME (Command Error)**  
Set to 1 when the command syntax is incorrect.  
Example Incorrectly spelled command name; "9" used in octal data.
- **Bit 4 EXE (Execution Error)**  
Set to 1 when the command syntax is correct but the command cannot be executed in the current state.  
Example Received a command with a parameter outside the range or attempted to output a hard copy while waveform acquisition is in progress.
- **Bit 3 DDE (Device Dependent Error)**  
Set to 1 when execution of the command is not possible due to an internal problem in the instrument that is not a command error or an execution error.
- **Bit 2 QYE (Query Error)**  
Set to 1 if the output queue is empty or if the data is missing even after a query has been sent.  
Example No response data; data is lost due to an overflow in the output queue.
- **Bit 1 RQC (Request Control)**  
Not used (always 0)
- **Bit 0 OPC (Operation Complete)**  
Set to 1 when the operation designated by the \*OPC command (see chapter 5) has been completed.

### Bit Masking

To mask a bit in the standard event register so that it does not cause bit 5 (ESB) of the status byte to change, set the corresponding bit in the standard event enable register to 0. Refer to Chapter 4.

For example, to mask bit 2 (QYE) so that ESB will not be set to 1, even if a query error occurs, set bit 2 of the standard event enable register to 0. This can be done using the \*ESE command. To inquire whether each bit of the standard event enable register is 1 or 0, use the \*ESE?. For details on the \*ESE command, see chapter 5.

### Operation of the Standard Event Register

The standard event register is provided for eight different kinds of event which can occur inside the instrument. Bit 5 (ESB) of the status byte is set to 1 when any of the bits in this register becomes 1 (or when the corresponding bit of the standard event enable register becomes 1).

Example

1. A query error occurs.
2. Bit 2 (QYE) is set to 1.
3. Bit 5 (ESB) of the status byte is set to 1 if bit 2 of the standard event enable register is 1.

It is also possible to check what type of event has occurred inside the instrument by reading the contents of the standard event register.

### Reading from the Standard Event Register

The contents of the standard event register can be read by the \*ESR command. After the register is read, it is cleared.

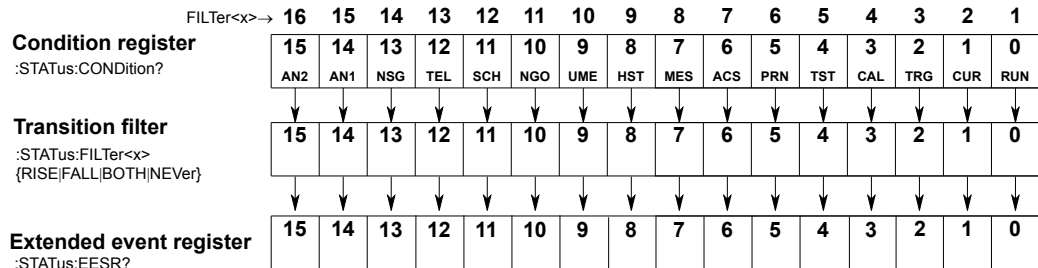
### Clearing the Standard Event Register

The standard event register is cleared in the following three cases.

- When the contents of the standard event register are read using the \*ESR command.
- When a \*CLS command is received.
- When the instrument is power cycled.

## 6.4 Extended Event Register

Reading the extended event register tells you whether changes in the condition register (reflecting internal conditions) have occurred. A filter can be applied which allows you to decide which events are reported to the extended event register.



The meaning of each bit of the condition register is as follows:

Bit 0	RUN (Running)	Set to 1 while waveform acquisition is in progress.
Bit 1	CUR (Cursor)	Set to 1 during cursor measurement.
Bit 2	TRG (Awaiting trigger)	Set to 1 when waiting for a trigger.
Bit 3	CAL (Calibration)	Set to 1 while calibration is in progress.
Bit 4	TST (Testing)	Set to 1 while self-test is in progress.
Bit 5	PRN (Printing)	Set to 1 while the built-in printer is operating, while data is being output to an external printer (USB/network), or while screen image data is being saved.
Bit 6	ACS (Accessing)	Set to 1 while a storage drive is being accessed.
Bit 7	MES (Measuring)	Set to 1 when automated measurement of waveform parameters is in progress.
Bit 8	HST (History Search)	Set to 1 while history search is in progress.
Bit 9	UME (User Math Executing)	Set to 1 while the user-defined computation is in progress.
Bit 10	NGO (Go/No-go)	Set to 1 while GO/NO-GO search is in progress.
Bit 11	SCH (Search)	Set to 1 while search is in progress.
Bit 12	TEL (Telecom Test)	Set to 1 while the telecom test is in progress.
Bit 13	NSG (N-Single)	Set to 1 while continuous acquisition is in progress when the trigger mode is set to single (N).
Bit 14	AN1 (Analysis1)	Set to 1 while Analysis 1 is in progress.
Bit 15	AN2 (Analysis2)	Set to 1 while Analysis 2 is in progress.

The transition filter parameters detect changes in the specified bit (numerical suffix, 1 to 16) of the condition register in the following manner and overwrite the extended event register.

RISE	The specified bit of the extended event register is set to 1 when the bit of the condition register changes from 1 to 0.	
FALL	The specified bit of the extended event register is set to 1 when the bit of the condition register changes from 0 to 1.	
BOTH	The bit of the extended event register is set to 1 when the bit of the condition register changes from 0 to 1 or from 1 to 0.	
NEVer	Always 0.	

## 6.5 Output Queue and Error Queue

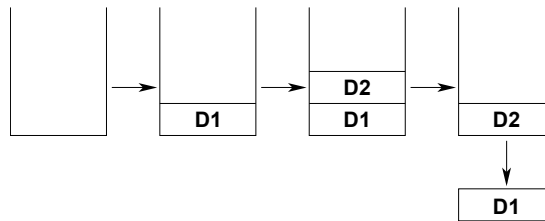
### Output Queue

The output queue is provided to store response messages to queries. For example, if you send the WAVEform:SEND? command, which requests the output of acquired data, the data is stored in the output queue until it is read.

As shown below, data are stored in order and read from the oldest ones first. The output queue is emptied in the following cases (in addition to when read-out is performed).

- When a new message is received from the controller.
- When a deadlock occurs (see page 5-2).
- When a device clear command (DCL or SDC) is received.
- When the instrument is power cycled.

The output queue cannot be emptied using the \*CLS command. To see whether the output queue is empty or not, check bit 4 (MAV) of the status byte.



### Error Queue

The error queue stores the error No. and message when an error occurs. For example, if the controller sends an incorrect program message, the error number and message “113, “Undefined header”” are stored in the error queue when the error is displayed.

The STATUS:ERRor? query can be used to read the contents of the error queue. As with the output queue, the messages are read from the oldest ones first.

When the error queue overflows, the last message is replaced by the message “350, “Queue overflow”.”

The error queue is also cleared for the following cases:

- When a \*CLS command is received.
- When the instrument is power cycled.

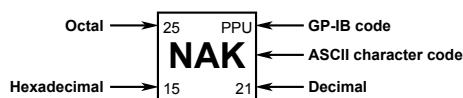
To see whether the error queue is empty or not, check bit 2 (EAV) of the status byte.

# Appendix 1 ASCII Character Codes

The following table shows the ASCII character codes.

	0	1	2	3	4	5	6	7
0	0 NUL	20 DEL	40 SP	60 0	100 @	120 P	140 '	160 p
1	1 SOH	21 DC1	41 !	61 1	101 A	121 Q	141 a	161 q
2	2 STX	22 DC2	42 "	62 2	102 B	122 R	142 b	162 r
3	3 ETX	23 DC3	43 #	63 3	103 C	123 S	143 c	163 s
4	4 EOT	24 DC4	44 \$	64 4	104 D	124 T	144 d	164 t
5	5 ENQ	25 NAK	45 %	65 5	105 E	125 U	145 e	165 u
6	6 ACK	26 SYN	46 &	66 6	106 F	126 V	146 f	166 v
7	7 BEL	27 ETB	47 ,	67 7	107 G	127 W	147 g	167 w
8	10 BS	30 CAN	50 (	70 8	110 H	130 X	150 h	170 x
9	11 HT	31 EM	51 )	71 9	111 I	131 Y	151 i	171 y
A	12 LF	32 SUB	52 *	72 :	112 J	132 Z	152 j	172 z
B	13 VT	33 ESC	53 +	73 ;	113 K	133 [	153 k	173 {
C	14 FF	34 FS	54 ,	74 <	114 L	134 \ 	154 l	174 
D	15 CR	35 GS	55 -	75 =	115 M	135 ] ]	155 m	175 }
E	16 SO	36 RS	56 .	76 >	116 N	136 ^	156 n	176 ~
F	17 SI	37 US	57 /	77 ?	117 O	137 _	157 o	177 DEL (RUBOUT)
	Address commands	Universal commands	Listener address		Talker address		Secondary commands	

### Example





## Appendix 2 Error Messages

This section describes the error messages related to communications.

- The messages can be displayed in English or Japanese on the DL9000. However, when the messages are read from a PC or other similar computers, the messages are displayed in English.
- If servicing is required, contact your nearest YOKOGAWA dealer for repairs.
- Only error messages related to communications are listed here. For other error messages, see *User's Manual IM 701310-01E*.

· Communication syntax error	100~199
· Communication execution error	200~299
· Model specific (other)	300~398
· Communication query error	400~499
· System error (communications)	399

} Details given below.

### Error in Communication Command (100-199)

Code	Messages	Corrective Action	Reference Page
102	Syntax error.	Invalid syntax.	Chapter 4, 5
103	Invalid separator.	Use a comma to separate the data.	4-1
104	Data type error.	Write using the correct data form.	4-6 to 4-7
105	GET not allowed.	GET is not supported for responses to interface messages.	3-6
108	Parameter not allowed.	Check the number of data points.	4-6, Chapter 5
109	Missing parameter.	Enter the required data.	4-6, Chapter 5
111	Header separator error.	Use a space to separate the header and data.	4-1
112	Program mnemonic too long.	Check the mnemonic (alphanumerical character string).	Chapter 5
113	Undefined header.	Check the header.	4-4, Chapter 5
114	Header suffix out of range.	Check the header.	4-4, Chapter 5
120	Numeric data error.	A number is required in the <NRf> form.	4-6
123	Exponent too large.	Use a smaller exponent for <NR3> format.	4-6, Chapter 5
124	Too many digits.	The value must be less than equal to 255 digits.	4-6, Chapter 5
128	Numeric data not allowed.	Enter in a format other than <NRf> format.	4-6, Chapter 5
131	Invalid suffix.	Check the unit of the <Voltage>, <Time>, <Frequency>, and <Current>.	4-6
134	Suffix too long.	Check the unit of the <Voltage>, <Time>, <Frequency>, and <Current>.	4-6
138	Suffix not allowed.	No units are allowed other than <Voltage>, <Time>, <Frequency>, and <Current>.	4-6
141	Invalid character data.	Select character data from the selections available in {... ... ...}.	4-7, Chapter 5
144	Character data too long.	Check the spelling of the character strings in {... ... ...}.	4-7, Chapter 5
148	Character data not allowed.	Write in a data form other than {... ... ...}.	4-5, Chapter 5
150	String data error.	Enclose <String> in double quotation or single quotation marks.	4-7
151	Invalid string data.	<String> is too long or contains characters which cannot be used.	4-7, Chapter 5
158	String data not allowed.	Enter in a data format other than <Character string>.	4-6, Chapter 5
161	Invalid block data.	<Block data> is not allowed.	4-7, Chapter 5
168	Block data not allowed.	<Block data> is not allowed.	4-7, Chapter 5
171	Invalid expression.	Equations cannot be used.	Chapter 5
178	Expression data not allowed.	Equations cannot be used.	Chapter 5

**Error in Communication Execution (200 to 299)**

Code	Messages	Corrective Action	Reference Page
221	Setting conflict.	Check the relevant settings.	Chapter 5
222	Data out of range.	Check the range.	Chapter 5
223	Too much data.	Check the length of the data.	Chapter 5
224	Illegal parameter value.	Check the range.	Chapter 5
241	Hardware missing.	Check the installed options.	–
260	Expression error.	Equations cannot be used.	–

**Error in Communication Query (400 to 499)**

Code	Messages	Corrective Action	Reference Page
410	Query INTERRUPTED.	Check transmission/reception order.	4-2
420	Query UNTERMINATED.	Check transmission/reception order.	4-2
430	Query DEADLOCKED.	Limit the length of the program message including <PMT> to 1024 bytes or less.	–
440	Query UNTERMINATED after indefinite response.	Do not specify a query after the *IDN? or *OPT? command.	–

**Error in System Operation (399)**

Code	Messages	Corrective Action	Reference Page
399	Fatal error in the communication driver.	Maintenance service is required.	–

**Warning (50)**

Code	Messages	Corrective Action	Reference Page
50	*OPC/? exists in message.	Place the *OPC or *OPC? command at the end of the program message.	–

**Other Errors (350)**

Code	Messages	Corrective Action	Reference Page
350	Queue overflow.	Read the error queue.	6-6

**Note**

Code 350 indicates overflow of error queue. This code is returned as a response to the STATUS:ERRor? query; it does not appear on the screen.

## Appendix 3 Waveform Parameter Name Table

Name Displayed on the Setup Menu of the DL9000 Screen	Name Used by Communication Commands	Name on the DL9000 Screen When Displaying Measured Results
Max	MAXimum	Max
Min	MINimum	Min
High	HIGH	High
Low	LOW	Low
P-P	PTOPeak	P-P
Hi-Low	HiLow	Hi-Low
+Over	POVershoot	+Over
-Over	NOVershoot	-Over
Rms	RMS	Rms
Mean	MEAN	Mean
Sdev	SDEViation	Sdev
IntegTY	TYINteg	ITY
C.Rms	CRMS	CRms
C.Mean	CMEan	CMean
C.Sdev	CSDeviation	CSdev
C.IntegTY	TYCInteg	CITY
Freq	FREQuency	Freq
1/Freq	PERFrequency	1/FR
Count	COUNT	Count
Burst	BURSt	Burst
+Width	PWIDth	+Width
-Width	NWIDth	-Width
Period	PERiod	Period
Duty	DUTYcycle	Duty
Rise	RISE	Rise
Fall	FALL	Fall
Delay	DELay	Dly

## Appendix 4 Eye Pattern Parameter Name Table

Name Displayed on the Setup Menu of the DL9000 Screen	Name Used by Communication Commands	Name on the DL9000 Screen When Displaying Measured Results
Crossing %	PCROssing	Cross%
Eye Height	EHEight	EyeHi
Eye Width	EWIDth	EyeWid
Q Factor	QFACTOR	QFact
Jitter	JITTer	Jitter
Duty Cycle Distraction %	PDUTYcycle	DCDTime%
Vtop	VTOP	Vtop
Vbase	VBASe	Vbase
$\sigma$ top	SDTop	$\sigma$ top
$\sigma$ base	SDBase	$\sigma$ base
Tcrossing1	T1CROSSing	Tcros1
Tcrossing2	T2CROSSing	Tcros2
Vcrossing	VCROSSing	Vcros
Ext Rate dB	DBERate	ERdB
Rise	RISE	Rise
Fall	FALL	Fall

# Index

Symbol	Page
<boolean>	4-7
<current>	4-6
<frequency>	4-6
<RMT>	4-1
<time>	4-6
<voltage>	4-6

A	Page
accumulated histogram	
automated measurement	5-36
automated measurement of waveform parameter	5-39
calculation item	5-37
cursor measurement	5-36
horizontal cursor	5-38
horizontal range	5-35, 5-42
measurement target window	5-42
source trace	5-42
vertical cursor	5-38
vertical range	5-42
accumulate mode	5-82
ACK condition	5-153
ACQUIRE group	5-34
acquisition mode	5-201
action-on-trigger	
action count	5-149
buzzer	5-149
mail transmission	5-149
output screen image data	5-149
save the waveform data	5-150
addition	5-119
address	3-5
analysis function display	5-42
analysis function type	5-52
ANALYSIS group	5-35
ASETup group	5-65
automated measurement	5-122
automated measurement of waveform parameter	
detection mode	5-124
threshold level	5-124
trace	5-126
auto naming (file name)	5-89
auto scaling	5-67
auto scroll	5-203
auto setup	5-65
average	5-34

B	Page
backlight	5-83
block data	4-7
brightness of the backlight	5-83

C	Page
CALibrate group	5-66
calibration	5-66
CAN bus signal trigger (enhanced)	5-177
CAN bus signal trigger (event)	5-153
CHANnel group	5-67
channel ON/OFF	5-67
character data	4-7
CLEAR group	5-69

click sound	5-142
clock channel (event trigger)	5-161
clock channel (trigger)	5-150
command	4-3
command group	4-3
common command	4-3
Common Command group	5-204
COMMunicate group	5-70
communication status function	5-140
computation	5-114
computation reference point	5-117
computation unit	5-120
condition register	5-140
continuous statistical processing	5-123
current directory	5-85
CURSor group	5-72
cursor ON/OFF	5-72
cycle statistical processing	5-123

D	Page
data	4-6
data rate (Ethernet)	2-2
data rate (GP-IB)	3-4
date	5-142
deadlock	4-2
delay computation	5-114
delay measurement	
edge detection count	5-128
polarity	5-128
determination mode (event interval)	5-175
directory	5-86
display format	5-83
DISPlay group	5-82
display of the reference	5-131
distal value	5-129

E	Page
edge/state trigger	5-162, 5-196
edge count computation	
edge detection polarity	5-114
hysteresis	5-114
enhanced trigger	5-176
error code	5-140
error message	App-2
error queue	6-6
event interval	5-151
extended event enable register	5-140
extended event register	6-5
external printer output	5-100
eye pattern	5-143
eye pattern parameter name	App-5

## Index

<b>F</b>	Page
<b>FFT</b>	
automated measurement	5-44
automated measurement mode	5-48
center point of magnification	5-51
horizontal center	5-43
horizontal mode	5-44
horizontal span	5-43
marker cursor measurement	5-45
maximum value hold function	5-44
measurement source window	5-51
peak value measurement	5-48
power value	5-46
source trace	5-44
vertical axis mode	5-51
vertical sensitivity	5-52
window function	5-52
FFT search	5-105
FILE group	5-85
file output	5-100
filter type	5-116
format	5-142
free disk space	5-85
<b>G</b>	Page
GO/NO-GO determination	
buzzer	5-91
mail transmission	5-91
screen image print	5-91
send mail	5-91
GONogo group	5-91
GP-IB card	3-2
GP-IB interface	3-3
graticule	5-83
grid	5-83
<b>H</b>	Page
HCOPY group	5-100
high resolution mode	5-34, 5-101
history function	
display mode	5-103
end number	5-102
highlight display mode	5-103
replay function	5-103
start number	5-102
target record	5-103
history function of the reference	5-110
HISTORY group	5-102
history search function	
mode	5-104
source trace	5-109
source window	5-110
hold off time	5-196
horizontal cursor	
calculation item	5-73
ON/OFF	5-73
source trace	5-74
voltage value	5-72
horizontal sync signal	5-195
horizontal zoom	5-202
<b>I</b>	Page
I2C bus trigger (enhanced)	5-184
I2C bus trigger (event)	5-162
IIR filter computation	5-115
IMAGe group	5-112
INITialize group	5-113
initial point	5-117
input coupling	5-67
input filter	5-67
integral computation	5-116
intensity	5-83
interface message	3-6
interleave	5-34
internal memory	5-86
interpolation	5-34, 5-84
interval time (event interval)	5-176
inverted display	5-67
<b>L</b>	Page
load	5-87
local	5-71
local lockout	5-70
<b>M</b>	Page
mapping	5-84
marker cursor	
calculation item	5-74
delta T measurement	5-75
delta V value	5-75
time measurement	5-75
time value	5-76
voltage measurement	5-76
voltage value	5-76
marker cursor position	5-75
mask test	5-145
MATH group	5-114
MEASure group	5-122
measure search	
trace	5-107
waveform parameter	5-107
message	4-1
message information	5-140
message language	5-142
minimum record number	5-201
mode (automated measurement)	5-123
model	5-204
moving average computation	5-115
multiplication	5-118
<b>N</b>	Page
NO-GO action	5-91
number of FFT point	5-44
<b>O</b>	Page
offset (MATH)	5-120
offset cancel	5-68
offset voltage	5-68
operation pending status register	5-70
operator	5-119
option	5-206
OR trigger	5-196
output of screen data	5-100
overlap command	4-8, 5-70
<b>P</b>	Page
persistence	5-82
polarity (edge/state trigger)	5-196
probe attenuation	5-68
probe attenuation (external trigger)	5-198
program message	4-1
proximal value	5-129

pulse width search .....	5-136
pulse width trigger .....	5-198
pulse width trigger (event) .....	5-174

<b>Q</b>	Page
query .....	4-4

<b>R</b>	Page
record length .....	5-34
rectangle search .....	5-108
REFeRence group .....	5-131
remote .....	5-71
Remote Control .....	2-5, 3-5
repetitive sampling .....	5-34
rescaling .....	5-115
response .....	4-5
response data .....	4-2
response message .....	4-1

<b>S</b>	Page
sample rate .....	5-148, 5-201
scale display .....	5-68
scaling mode .....	5-120
search function	
clock channel .....	5-132
clock trace .....	5-132
detection count .....	5-133
detection waveform number .....	5-133
hold off detection .....	5-133
polarity of the clock channel .....	5-132
search logic .....	5-133
skip mode .....	5-133
SEARCh group .....	5-132
search source trace .....	5-135
search start position .....	5-135
search type .....	5-136
sequential command .....	4-8
serial cursor	
active level .....	5-76
bit length .....	5-77
bit rate .....	5-77
cursor type .....	5-78
display format .....	5-77
hysteresis .....	5-77
measured value .....	5-78
trace .....	5-78
serial cursor position .....	5-77
serial pattern search	
bit rate .....	5-135
chip select .....	5-134
clock .....	5-134
latch .....	5-134
serial pattern trigger (enhanced) .....	5-189
serial pattern trigger (event) .....	5-169
serial poll .....	5-140
skew correction .....	5-67
skip mode .....	5-133
SNAP group .....	5-137
SPI bus trigger (enhanced) .....	5-191
SPI bus trigger (event) .....	5-171
SStArt group .....	5-138
standard event register .....	6-4
StARt group .....	5-139
statistical processing (history data) .....	5-123
status byte .....	6-3
STATus group .....	5-140
status report .....	6-1

STOP group .....	5-141
string data .....	4-7
subtraction .....	5-117
SYSTem group .....	5-142
system information .....	5-142

<b>T</b>	Page
T/div value .....	5-148
target record number .....	5-200
target waveform .....	5-201
telecom test	
condition .....	5-92
eye pattern .....	5-93, 5-143
mask determination .....	5-93
mask test .....	5-145
measurement range .....	5-147
source trace .....	5-147
type .....	5-143
TELecomtest group .....	5-143
time .....	5-142
TIMEbase group .....	5-148
total number .....	5-200
transition filter .....	5-140
trial mode .....	5-176
trigger delay .....	5-150
TRIGger group .....	5-149
trigger level (external trigger) .....	5-198
trigger mode .....	5-196
trigger position .....	5-196, 5-201
trigger source .....	5-197
trigger source (edge/state) .....	5-196
trigger source logic .....	5-198
trigger type .....	5-198
trigger type of the event .....	5-174
TV trigger .....	5-193

<b>U</b>	Page
upper-level query .....	4-4
USB interface .....	1-2
USB interface connector .....	1-1

<b>V</b>	Page
V/div .....	5-68
vertical cursor	
1/delta T value .....	5-79
calculation item .....	5-80
delta T value .....	5-78
time value .....	5-79
vertical cursor position .....	5-79
vertical position .....	5-68
vertical sensitivity .....	5-68
vertical zoom .....	5-203
VT cursor	
calculation item .....	5-81
time value .....	5-80
voltage value .....	5-81
VT cursor position .....	5-80
VT waveform display .....	5-52

<b>W</b>	Page
waveform acquisition mode .....	5-34
waveform display color .....	5-83
WAVeform group .....	5-200
waveform label .....	5-67
waveform mapping .....	5-84

## Index

---

waveform parameter	
automated measured value.....	5-128
statistical value.....	5-127
waveform parameter measurement function	
histogram.....	5-53
list display.....	5-56
mode.....	5-57
trace.....	5-57
trend display.....	5-57
waveform parameter name.....	App-4
waveform zoom.....	5-202
wave search	
zone edit menu.....	5-109
window trigger width.....	5-198

## **X** Page

---

X-axis trace.....	5-64
XY display function	
automated measurement.....	5-61
cursor measurement.....	5-61
gate function.....	5-60
integration.....	5-63
integration polarity.....	5-63
measurement range.....	5-63
measurement source window.....	5-63
XY search.....	5-108

## **Y** Page

---

Y-axis trace.....	5-64
-------------------	------

## **Z** Page

---

zone/parameter determination.....	5-94
ZOOM group.....	5-202
zoom type.....	5-203