

AQ7277 Remote OTDR Communication Interface

Foreward

Thank you for purchasing YOKOGAWA's AQ7277.

This Communication Interface User's Manual describes the functions and commands of USB and Ethernet interface. To ensure proper use of the USB/Ethernet interfaces, please read this manual thoroughly. Keep the manual in a safe place for quick reference whenever a question arises.

Notes

- The contents of this manual are subject to change without prior notice as a result of improvements in 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 representative.
- Copying or reproduction of all or any part of the contents of this manual without YOKOGAWA's permission is strictly prohibited.

Trademarks

- Microsoft, Internet Explorer, Windows, and WindowsVista are either registered trademarks or trademarks of Microsoft Corporation in the United States and/or other countries.
- Adobe, Acrobat, and PostScript are either trademarks or registered trademarks of Adobe Systems incorporated.
- For purposes of this manual, the ® and TM symbols do not accompany their respective registered trademark names or trademark names.
- Other product names are registered trademarks or trademarks of their respective holders.

Revisions

1st Edition:	March 2011
2nd Edition:	October 2017

How to Use this Manual

Structure of the Manual

This User's Manual consists of the following sections:

Chapter 1 Remote Control Function Overview

Gives an overview of the various communication interfaces.

Chapter 2 Connecting the PC

Describes the functions and specifications of the USB interface and the Ethernet Interface used to control the AQ7277 from a PC.

Chapter 3 Before Programming

Describes the syntax used to transmit commands.

Chapter 4 Remote Commands

Describes each command that is available.

Conventions Used in This Manual

Notations Used in the Procedural Explanations

On pages that describe the operating procedures in each chapter, the following notations are used to distinguish the procedure from their explanations.

Procedure

This subsection contains the operating procedure used to carry out the function described in the current section. The procedures are written with inexperienced users in mind; experienced users may not need to carry out all the steps.

Explanation

This subsection describes the setup parameters and the limitations on the procedures.

Note

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

Units

Symbol	Description	Example
k	1000	400km
K	1024	459KB (file size)

Symbols Used in Syntax Descriptions

Symbols which are used in the syntax descriptions in Chapter 4 are shown below. These symbols are referred to as BNF notation (Backus-Naur Form).

For detailed information, see section 3.4, "Data."

Symbol	Description	Example	Example of Input
<>	Defined value	SET:M<x> <x> = 1,2,3	-> SET:M2
{ }	One of the options in {} is selected.	LMTechnique {LSA TPA}	-> LMTechnique TPA
	Exclusive OR		

Index

1

How to Use this Manual	ii
Conventions Used in This Manual	iii

2

Chapter 1 Remote Control Function Overview

1.1 Remote Interface	1-1
1.2 Switching between Remote and Local Modes	1-2

3

Chapter 2 Connecting the PC

2.1 Connecting via the USB	2-1
2.2 Connecting via the Ethernet Interface	2-2

4

Chapter 3 Before Programming

3.1 Messages	3-1
3.2 Commands	3-3
3.3 Response	3-6
3.4 Data	3-7
3.5 Synchronization with the Controller	3-11

Chapter 4 Commands

4.1 A List of Commands	4-1
4.2 Common Commands	4-4
4.3 Device-Specific Commands	4-6
ACQuire Group	4-6
ANALysis Group	4-9
COMMunicate Group	4-13
DISPlay Group	4-14
FILE Group	4-16
MENU Group	4-16
MISC Group	4-17
NETWork Group	4-18
SETup Group	4-19
STATus Group	4-19
WAVedata Group	4-20
SYSTem Group	4-21
4.4 Condition Register	4-22
4.5 Output Queue and Error Queue	4-23
4.6 Error Messages	4-24

1.1 Remote Interface

The AQ7277 is equipped with the following remote interfaces.

USB 1.1 Interface (Type B. See Chapter 2)

This interface is used by a controller such as a PC to remotely control the AQ7277.

A controller is connected to this interface.

Remote commands are used to control the AQ7277.

Dedicated commands for the AQ7277 complying with SCPI (Standard Commands for Programmable Instruments) are available for the remote commands (see chapter 4).

Ethernet Interface (See Chapter 2)

This interface is used by a controller such as a PC to remotely control the AQ7277 via the network.

Downloading the Library and Driver

The items below are needed on the PC to use the communication functions via the USB interface.

- Communication Library(TMCTL)
- USB connection device driver between the PC and AQ7277(YKUSB)

The items below are needed on the PC to use the communication functions via the Ethernet interface.

- Communication Library(TMCTL)

The library and driver above can be downloaded from the following Web page.

http://www.yokogawa.com/tm/dl_driver.htm

1.2 Switching between Remote and Local Modes

When Switching from Local to Remote Mode

The AQ7277 is in local mode when it is being controlled with the remote controller for the AQ7277 Remote OTDR. It is in remote mode when it is being controlled through the use of communication commands.

In remote mode, the soft keys on the remote controller for the AQ7277 Remote OTDR are disabled.

- All keys except the "Local" key are disabled.
- Settings entered in local mode are passed on even when the AQ7277 switches to remote mode.

When Switching from Remote to Local Mode

In remote mode, if you press Local on the remote controller for the AQ7277 Remote OTDR, the AQ7277 switches to local mode.

- You will be able to use the remote controller for the AQ7277 Remote OTDR.
- Settings entered in remote mode are retained even when the AQ7277 switches to local mode.

2.1 Connecting via the USB

You can control the AQ7277 from a PC using the USB interface.

USB Interface Functions and Specifications

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 data, and error codes.

Transmission Function

Outputs measured and computed data.

Outputs setup data and the status byte.

Outputs error codes that have occurred.

USB Interface Specifications

Electrical and mechanical specifications:	Conforms to USB Rev.1.0
Connector:	Type B connector (receptacle)
Number of ports:	1
Power supply:	Self-powered
Compatible PC systems:	PCs with standard USB ports running Windows XP, Windows Vista, or Windows 7. (A separate device driver is required to connect to a PC.)

Connection Procedure

Connecting a USB Cable to Remotely Control the AQ7277

Connect a USB cable to the Type B connector.

Precautions to Be Taken When Connecting the Cable

- Connect the USB cable by inserting the connector firmly into the USB connector.
- Do not connect or disconnect the USB cable after the power is turned ON until the AQ7277 is ready for operation (approximately 20 s).

2.2 Connecting via the Ethernet Interface

You can control the AQ7277 from a PC by connecting the AQ7277 to a LAN using the Ethernet interface.

Ethernet Interface Functions and Specifications

Reception Function

You can specify the same settings as those specified by AQ7277 Remote Controller.
Receives output requests for measured and computed data, setup data, and error codes.

Transmission Function

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

Ethernet Interface Specifications

Number of communication ports:	1
Electrical and mechanical specifications:	Conforms to IEEE802.3
Transmission system:	Ethernet (10BASE-T/100BASE-TX)
Data rate:	10 Mbps/100 Mbps
Communication protocol:	TCP/IP
Connector type:	RJ45 connector
Port number used:	10001/tcp

Note

When using the Ethernet interface, a user name and password are required when connecting to the network. For the procedure to enter the user name and password, see the User's Manual of AQ7277 Remote Controller

Connection Procedure

Connect a UTP (Unshielded Twisted-Pair) cable or an STP (Shielded Twisted-Pair) cable that is connected to a hub, for example, to the 100BASE-TX port.

Precautions to Be Taken When Connecting the Cable

- Be sure to use a straight cable via a hub for the connection between the AQ7277 and the PC.
- When using a UTP cable (straight cable), use a cable of category 5.

Note

Set the TCP/IP address using the remote controller for the AQ7277 Remote OTDR. For the procedure, see the user's manual for the remote controller for the AQ7277 Remote OTDR.

3.1 Messages

Blocks of message data are transferred between the controller and this instrument during communications. Messages sent from the controller to this instrument are called program messages, and messages sent back from this instrument to the controller are called response messages.

If a program message contains a query command, i.e. a command which requests a response, this instrument returns a response message. A single response message is always returned in reply to a program message.

Program Messages

Program Message Unit

A program message consists of one or more program message units; each unit corresponds to one command. This instrument executes commands one by one according to the order in which they are received.

Program message units are delimited by a ";".

"For a description of the format of the program message unit, refer to the explanation given further below.

Example `:ACQuire:AVERage:MODE HIREFLECTION;TIME AUTO<PMT>`

<PMT>

PMT is a terminator used to terminate each program message. The following three types of terminator are available.

- **NL (New Line)**
Same as LF (Line Feed). ASCII code "0AH" is used.
- **^END**
END message defined in IEEE488.1. (EOI signal)
(The data byte sent with an END message will be the final item of the program message unit.)
- **NL^END**
NL with an END message attached (NL is not included in the program message unit.)

<Program Header>

A program header is used to indicate the command type. For details, refer to page 3-3.

<Program Data>

If certain conditions are required for the execution of a command, program data must be added. Program data must be separated from the header by a space (ASCII code "20H"). If multiple items of program data are included, they must be separated by a ",", (comma). For details, refer to page 3-5.

Example `:ACQuire:AVERage:MODE HIREFLECTION<PMT>`

Response Messages

<Response Message Units>

A response message consists of one or more response message units: each response message unit corresponds to one response. Response message units are delimited by a ";" (semicolon).

Example :ACQuire:AVERage:MODE HIREFLECTION;TIME AUTO<PMT>

<RMT>

RMT is the terminator used for every response message. Only one type of response message is available; NL^END.

<Response Header>

A response header sometimes precedes the response data. Response data must be separated from the header by a space. For details, refer to page 3-5.

<Response Data>

Response data is used to define a response. If multiple items of response data are used, they must be separated by a "," (comma). For details, refer to page 3-5.

Example 1200<RMT> :ACQuire:AVERage:MODE HIREFLECTION<PMT>

If a program message contains more than one query, responses are made in the same order as the queries. Normally, each query returns only one response message unit, but there are some queries which return more than one response message unit. The first response message unit always responds to the first query, but it is not always true that the 'n'th unit always responds to the 'n'th query. Therefore, if you want to make sure that a response is made to each query, the program message must be divided up into individual messages.

Points to Note Concerning Message Transmission

- It is always possible to send a program message if the previous message which was sent did not contain any queries.
- If the previous message contained a query, it is not possible to send another program message until a response message has been received. An error will occur if a program message is sent before a response message has been received in its entirety. A response message which has not been received will be discarded.
- If an attempt is made by the controller to receive a response message, even if there is no response message, an error will occur. An error will also occur if the controller makes an attempt to receive a response message before transmission of a program message has been completed.
- If a program message of more than one unit is sent and some of the units are incomplete, this instrument receives program message units which the instrument thinks complete and attempts to execute them. However, these attempts may not always be successful and a response may not always be returned, even if the program message contains queries.

3.2 Commands

There are two types of command (program header) which can be sent from the controller to this instrument. They differ in the format of their program headers.

Common command header

Common Command Header Commands defined in IEEE 488.2-1987 are called common commands.

An asterisk (*) must always be attached to the beginning of a command.

An example of a common command

*CLS

Compound header

Compound Header Commands designed to be used only with this instrument are classified and arranged in a hierarchy according to their function. The format of a compound header is below. A colon (:) must be used when specifying a lower-level header.

An example of a compound header

: ACQuire: AVERage: MODE

When Concatenating Commands

Command Group

A command group is a group of commands which have the same compound header. A command group may contain sub-groups.

Example Commands relating to acquisition settings

```
:ACQuire:AVERage:MODE
:ACQuire:ATTenuation
:ACQuire:AVERage:TYPE
:ACQuire:DRANge
:ACQuire:PWIDth
:ACQuire:REALtime:START
:ACQuire:SETTing
```

When Concatenating Commands of the Same Group

This instrument stores the hierarchical level of the command which is currently being executed, and performs analysis on the assumption that the next command to be sent will also belong to the same level. Therefore, it is possible to omit the header if the commands belong to the same group.

Example

```
:ACQuire:AVERage:MODE HIREFLECTION;TIME AUTO<PMT>
```

When Concatenating Commands of Different Groups

A colon (:) must be included before the header of a command, if the command does not belong to the same group as the preceding command. It is impossible to omit a colon (:).

Example

```
:ACQuire:AVERage:MODE HIREFLECTION;:DISPlay:CURSor
:SECond ON<PMT>
```

When Concatenating Common Commands

Common commands defined in IEEE 488.2-1987 are independent of hierarchical level. Thus, it is not necessary to add a colon (:) before a common command.

Example

```
:ACQuire:AVERage:MODE HIREFLECTION;*CLS;TIME AUTO<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 typed in for each command even when commands of the same command group are being concatenated.

Example

```
:ACQuire:AVERage:MODE HIREFLECTION<PMT>;:ACQuire:AVERage
:TIME AUTO<PMT>
```

Upper-level Query

An upper-level query is a compound header to which a question mark is appended. Execution of an upper-level query allows all a group's settings to be output at once. Some query groups comprising more than three hierarchical levels can output all their lower level settings.

Example

```
:ANALysis:THReshold?<PMT>
->:ANAL:THR:EOF 3.0;FERL 40.0;FESL 1.00;RLOS 70.0
;SLOS 0.03
```

Note

- In reply to a query, a response can be returned as a program message to this instrument. Transmitting a response can restore the settings made when the query was executed.
- Not all a group's information will necessarily be sent out as a response. Some upper-level queries will not return setup data which is not currently in use.

Header Interpretation Rules

This instrument interprets the header received according to the following rules.

- Mnemonics are not case sensitive.

Example

"DRANge" can also be written as "drange" or "Drange."

- The lowercase part of a header can be omitted.

Example

"DRANge" can also be written as "DRANG" or "DRAN."

- If the header ends with a question mark, the command is a query. It is not possible to omit the question mark.

Example

"DRANge?" cannot be abbreviated to anything shorter than "DRAN?."

- If the "x" at the end of a mnemonic is omitted, it is assumed to be "1."

Example If "M<x>" is written as "M," this represents "M1."

Note

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

3.3 Response

Form

On receiving a query from the controller, this instrument returns a response message to the controller. A response message is sent in one of the following two forms.

Response Consisting of a Header and Data

If the query can be used as a program message without any change, a command header is attached to the query, which is then returned.

Example: ACQUIre:MODE?<PMT> ->:ACQUIre:MODE NORMAL<RMT>

Response Consisting of Data Only

If the query cannot be used as a program message unless changes are made to it (i.e. it is a query-only command), no header is attached and only the data is returned. Some query-only commands can be returned after a header is attached to them.

Example: MEASure:CHANnel1:PTOPeak:VALue?<PMT> -> 10.0E+00<RMT>

When Returning a Response without a Header

It is possible to remove the header from a response consisting of a header and data. The "COMMunicate:HEADer" command is used to do this.

Abbreviated Form

Normally, the lowercase part 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 part enclosed by [] is also omitted in the abbreviated form.

3.4 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	Description
<Decimal>	Value expressed as a decimal number (Example: Average time -> ACQuire:AVERage:TIME 60)
<Distance><Time> <Wavelength><Loss>	Physical value (Example: Distance range -> ACQuire:DRANge 500)
<Register>	Register value expressed as either binary,octal, decimalor hexadecimal (Example: Extended event register value -> STATus:EESE #HFE)
<Character data>	Specified character string (mnemonic). Can be selected from { } (Example: Setup mode -> ACQuire:SETTing{SIMPLE DETAIL WIZARD MULTI})
<Boolean>	Indicates ON/OFF. Set to ON or OFF (Example: Fault event display -> ANALysis:FEDetection ON)
<Character string data>	Arbitrary character string (Example: Comment to be saved -> FILE:SAVe:COMMeNt "ABCDEF")
<Filename>	Gives the name of a file. (Example: Name of file to be saved -> FILE:SAVe:WAVeform:NAME "CASE1")
<Block data>	Arbitrary 8-bit data (Example: Response to acquired waveform data -> #800000010ABCDEFGHJIJ)

<Decimal>

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

Symbol	Description	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.	

- Decimal values which are sent from the controller to this instrument can be sent in any of the forms to <NR3>. In this case, <NRf> appears.
- For response messages which are returned from this instrument to the controller, the form (<NR1> to <NR3> to be used) is determined by the query. The same form is used, irrespective of whether the value is large or small.
- In the case of <NR3>, the "+" after the "E" can be omitted, but the "-" cannot.
- If a value outside the setting range is entered, the value will be normalized so that it is just inside the range.
- If the value has more than the significant number of digits, the value will be rounded.

<Distance>, <Time>, <Wavelength>, and <Loss>

<Distance>, <Time>, <Wavelength>, and <Loss> indicate decimal values which have physical significance. <Multiplier> or <Unit> can be attached to <NRf>. They can be entered in any of the following forms.

Form	Example
<NRf><Multiplier><Unit>	0.85UM
<NRf><Unit>	500m
<NRf><Multiplier>	5M
<NRf>	5E -3

<Multiplier>

Multipliers which can be used are shown below.

Symbol	Word	Description
EX	Exa	10^{18}
PE	Peta	10^{15}
T	Tera	10^{12}
G	Giga	10^9
MA	Mega	10^6
K	Kilo	10^3
M	Milli	10^{-3}
U	Micro	10^{-6}
N	Nano	10^{-9}
P	Pico	10^{-12}
F	Femto	10^{-15}
A	Atto	10^{-18}

<Unit>

Units which can be used are shown below.

Symbol	Word	Description
M	Meter	Distance
S	Second	Time
dB	Decibel	Level
UM	Micro meter	Wavelength

- <Multiplier> and <Unit> are not case sensitive.
- "U" is used to indicate "μ."
- "MA" is used for Mega (M) to distinguish it from Milli.
- If both <Multiplier> and <Unit> are omitted, the default unit will be used.

<Register>

<Register> indicates an integer, and can be expressed in hexadecimal, octal, or binary as well as a decimal number. <Register> is used when each bit of a value has a particular meaning. <Register> 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 3-5.

Form	Example
{SIMPLE DETAIL WIZARD MULTI}	DETAIL

- As with a header, the "COMMunicate:VERBoSe"command can be used to return a response message in its full form. Alternatively, the abbreviated form can be used.
- The "COMMunicate:HEADer" command does not affect <character data>.

<Boolean>

<Boolean> is data which indicates ON or OFF, and is expressed in one of the following forms.

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

- When <Boolean> is expressed in <NRf> form, OFF is selected if the rounded integer value is "0" and ON is selected if the rounded integer is "Not 0."
- A response message is always "1" if the value is ON and "0" if it is OFF.

<Character String Data>

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

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

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

<Filename>

Gives the name of a file. The format is as follows.

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

- If you input an <NRf> value, the system converts the value (after rounding to the nearest integer) to the corresponding 8-character ASCII string. (If you set the value to 1, the name becomes "00000001".) Note that negative values are not allowed.
- The first 12 characters of <Character Data> are assigned as the file name.
- The first 14 characters of <Character String Data> are assigned as the file name.
- Response messages always return filenames as <character string> arguments.

<Block Data>

<Block data> is arbitrary 8-bit data. <Block data> is only used for response messages. Response messages are expressed in the following form.

Form	Example
#N<N-digit decimal value><Data byte string>	#800000010ABCDEFGH IJ

- #N
Indicates that the data is <Block data>. "N" is an ASCII character string number (digits) which indicates the number of data bytes that follow.
- <N-digits decimal value>
Indicates the number of bytes of data. (Example: 00000010=10 bytes)
- <Data byte string>
The actual data. (Example: ABCDEFGH IJ)
- 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.

3.5 Synchronization with the Controller

Achieving Synchronization

If you send the following program message when receiving averaging measurement data, the data may be retrieved before the completion of the measurement.

```
ACQuire: AVERage: STArT; : WAVedata: DISPlay: SEND: ASCii?
```

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

Using the STATus:CONDition? Query

The "STATus:CONDition?" command is used to query the contents of the condition register.

Whether averaging measurement waveforms are being retrieved can be determined by reading bit 1 of the condition register.

If bit 1 of the condition register is "1", waveforms are being retrieved. Otherwise, it is stopped. (Bit 1 is "0".)

Note

For details on the condition register, see section 4.4, "Condition Register."

List of Commands

Common Commands

Command	Function	Page
*CLS(Clear Status)	Clears all event status registers, the summary of which is reflected in the status byte register.	4-4
*ESE(Standard Event Status Enable)	Sets/queries the standard event enable register.	4-4
*ESR?(Standard Event Status Register)	Queries the standard event status register and simultaneously clears it.	4-4
*IDN? (Identification)	Queries the instrument type and firmware version.	4-4
*RST (Reset)	Executes a device reset to return the instrument to the known (default) status.	4-4
*SRE(Service Request Enable)	Sets/queries the service request enable register.	4-4
*STB?(Read Status Byte)	Queries the current value of the status byte register.	4-5
*TST?(Self Test)	Performs the instrument's self-test and queries the status.	4-5

Device-Specific Commands

Command	Function	Page
ACQUIRE Group		
:ACQUIRE:ATTenuation	Sets the attenuation or queries the current setting.	4-6
:ACQUIRE:AUTO:ATTenuation?	Queries the attenuation for AUTO.	4-6
:ACQUIRE:AUTO:DRANge?	Queries the distance range for AUTO.	4-6
:ACQUIRE:AUTO:PWIDth?	Queries the pulse width for AUTO.	4-6
:ACQUIRE:AVERage:COUNT?	Queries the current average count.	4-6
:ACQUIRE:AVERage:INDEX	Sets the average count or queries the current setting.	4-6
:ACQUIRE:AVERage:MODE	Sets the average mode or queries the current setting.	4-6
:ACQUIRE:AVERage:START	Executes the averaging measurement.	4-6
:ACQUIRE:AVERage:STOP	Stops the averaging measurement.	4-7
:ACQUIRE:AVERage:TIME	Sets the average time or queries the current setting.	4-7
:ACQUIRE:AVERage:TYPE	Sets the average unit or queries the current setting.	4-7
:ACQUIRE:DRANge	Sets the distance range or queries the current setting.	4-7
:ACQUIRE:OFFSet	Sets the horizontal measurement start position or queries the current setting.	4-7
:ACQUIRE:PWIDth	Sets the pulse width or queries the current setting.	4-8
:ACQUIRE:REALtime:START	Executes the realtime measurement.	4-8
:ACQUIRE:REALtime:STOP	Stops the realtime measurement.	4-8
:ACQUIRE:SMPinterval:DATA	Sets the sampling interval or queries the current setting.	4-8
:ACQUIRE:SMPinterval:VALue?	Queries the sampling interval.	4-8
:ACQUIRE:WAVelength	Sets the measured wavelength or queries the current setting.	4-8
:ACQUIRE:SELection:DRANge?	Queris the selectable distance range.	4-9
:ACQUIRE:SELection:WAVelength?	Queris the selectable measured wavelength.	4-9
:ACQUIRE:SELection:PWIDth?	Queris the selectable pulse width.	4-9
:ACQUIRE:SELection:ATTenuation?	Queris the selectable attenuatio.	4-9
:ACQUIRE:SELection:SMPinterval?	Queris the selectable sampling interval.	4-9
ANALYSIS Group		
:ANALYSIS:ASEarch:EXECute	Executes auto detection.	4-9
:ANALYSIS:ASEarch:NUMber?	Queries the number of auto detection events.	4-9
:ANALYSIS:BCOefficient	Sets the backscattering light level of the current wavelength or queries the current setting.	4-9
:ANALYSIS:IOR	Sets the group refraction index of the current wavelength or queries the current setting.	4-9
:ANALYSIS:CURSor:DELeTe	Clears the cursor.	4-9
:ANALYSIS:CURSor:DISTance	Sets the cursor position or queries the current setting.	4-9
:ANALYSIS:CURSor:DECibel?	Queries the cursor dB (decibel).	4-10
:ANALYSIS:CURSor:LINK	Sets the cursor link or queries the cursor link setting.	4-10
:ANALYSIS:EMARker:LMTechnique	Sets the approximation method (event) or queries the current setting.	4-10
:ANALYSIS:EMARker:SET:M1	Sets marker M1 of the current event or queries the current setting.	4-10
:ANALYSIS:EMARker:SET:M2	Sets marker M2 of the current event or queries the current setting.	4-10
:ANALYSIS:EMARker:SET:M3	Sets marker M3 of the current event or queries the current setting.	4-10

4.1 A List of Commands

Command	Function	Page
:ANALysis:EMARker:SET:Y2	Sets marker Y2 of the current event or queries the current setting.	4-10
:ANALysis:DUNit	Sets the distance unit or queries the current setting.	4-10
:ANALysis:EVENT:CURRENT:INDEX	Changes the current event.	4-10
:ANALysis:EVENT:CURRENT:NOTE	Sets the event note or queries the current setting.	4-10
:ANALysis:EVENT:IOR	Sets the section IOR of the current event or queries the current setting.	4-11
:ANALysis:EVENT:CURRENT:PEAK?	Retrieves the peak decibel of the current event.	4-11
:ANALysis:EVENT:FIX	Event Fix	4-11
:ANALysis:EVENT:CURRENT:DISTance?	Retrieves the distance of the current event.	4-11
:ANALysis:EVENT:CURRENT:LOSS?	Retrieves the splice loss of the current event.	4-11
:ANALysis:EVENT:CURRENT:RETurnloss?	Retrieves the return loss of the current event.	4-11
:ANALysis:EVENT:CURRENT:CUMLoss?	Retrieves the accumulated loss of the current event.	4-11
:ANALysis:EVENT:CURRENT:UNITloss?	Retrieves the loss per unit (dB/m) of the current event.	4-11
:ANALysis:EVENT:CURRENT:TYPE?	Retrieves the event type of the current event.	4-11
:ANALysis:EVENT:CURRENT:IOR?	Retrieves the section IOR of the current event.	4-11
:ANALysis:EVENT:DELeTe	Deletes the current event.	4-11
:ANALysis:EVENT:INSert	Inserts the event at the cursor position.	4-11
:ANALysis:FEDetection	Sets the fault event display or queries the current setting.	4-11
:ANALysis:FMARKer:DELeTe	Deletes the marker.	4-12
:ANALysis:FMARKer:LMTechnique	Sets the approximation method of the marker or queries the current setting.	4-12
:ANALysi s:FMARKer:LOSS?	Queries the splice loss.	4-12
:ANALysis:FMARKer:RETurnloss:VALue?	Queries the return loss.	4-12
	Queries the saturation of the return loss.	4-12
:ANALysis:FMARKer:REFLection:VALue?	Queries the reflection level.	4-12
	Queries the saturation of the reflection level.	4-12
:ANALysis:FMARKer:LEFT:LOSS?	Queries the loss between markers 1 and 2.	4-12
:ANALysis:FMARKer:LEFT:DISTance?	Queries the distance between markers 1 and 2.	4-12
:ANALysis:FMARKer:LEFT:UNITloss?	Queries the slope between markers 1 and 2.	4-12
:ANALysis:FMARKer:RIGHT:LOSS?	Queries the loss between markers 2 and 3.	4-12
:ANALysis:FMARKer:RIGHT:DISTance?	Queries the distance between markers 2 and 3.	4-12
:ANALysis:FMARKer:RIGHT:UNITloss?	Queries the slope between markers 2 and 3.	4-12
:ANALysis:FMARKer:SET:M<x>	Sets the marker or queries the current setting.	4-12
:ANALysis:FMARKer:SET:Y<x>	Sets the auxiliary marker or queries the current setting.	4-12
:ANALysis:REFEreNce:DELeTe	Deletes the distance reference.	4-13
:ANALysis:REFEreNce:DISTance	Sets the distance reference or queries the current setting.	4-13
:ANALysis:THREshold:EOFiber	Sets the threshold level of the end of fiber or queries the current setting.	4-13
:ANALysis:THREshold:RLOSSs	Sets the threshold level of the return loss or queries the current setting.	4-13
:ANALysis:THREshold:SLOSSs	Sets the threshold level of the splice loss or queries the current setting.	4-13
:ANALysis:TRACefix:STATe	Sets the tracefix or queries the current setting.	4-13
COMMunicate Group		
:COMMunicate?	Queries all settings related to communications.	4-13
:COMMunicate:HEADer	Sets whether to include a header in the response to a query or queries the current setting.	4-13
:COMMunicate:VERBoSe	Sets whether to return the response in full or abbreviated form or queries the current setting.	4-13
DISPlay Group		
:DISPlay:ALine	Sets the display of the approximation line or queries the current setting.	4-14
:DISPlay:COLor	Sets the screen color or queries the current setting.	4-14
:DISPlay:CURSor:DBValue	Sets the dB value of cursor or queries the current setting..	4-14
:DISPlay:CURSor:SECond	Sets the secondary cursor display or queries the current setting.	4-14
:DISPlay:CURSor:TYPE	Sets the cursor type or queries the current setting.	4-14
:DISPlay:DECibel:UPPer	Sets the display start level or queries the current setting.	4-14
:DISPlay:DIGit:DECibel	Sets dB display digit or queries the current setting.	4-14
:DISPlay:DIGit:DISTance	Sets the distance display unit or queries the current setting.	4-14
:DISPlay:DISTance:LEFT	Sets the display start distance or queries the current setting.	4-14
:DISPlay:DIVide:DECibel	Sets the vertical zoom rate or queries the current setting.	4-15
:DISPlay:DIVide:DISTance	Sets the horizontal zoom rate or queries the current setting.	4-15
:DISPlay:GTYPe	Sets grid display or queries the current setting.	4-15
:DISPlay:IMARk	Sets the marker information display or queries the current setting.	4-15
:DISPlay:ISCaLe	Initializes the display scale.	4-15
:DISPlay:OVERview	Sets overview display or queries the current setting.	4-15
:DISPlay:WAVE:TYPE	Sets the waveform type display format or queries the current setting.	4-15

Command	Function	Page
FILE Group		
:FILE:CSV:EVENT:VALId	Confirms if it is possible to make the csv format of the event list.	4-16
:FILE:CSV:EVENT:GET	Gets the CSV format of the event list.	4-16
:FILE:CSV:TRACe:VALId	Confirms if it is possible to make the csv format of the trace data.	4-16
:FILE:CSV:TRACe:GET	Gets the CSV format of the trace data.	4-16
:FILE:SOR:VALId?	Confirms if it is possible to make the SOR format of the trace data.	4-16
:FILE:SOR:GET?	Gets the sor format of trace.	4-16
MENU Group		
:MENU:ERRor:CLEAr	Deleting the error dialog display	4-16
:MENU:FUNCTion	Sets the function mode or queries the current setting.	4-16
:MENU:MARKer	Sets the marker mode or queries the current setting.	4-16
MISC Group		
:MISC:DATE:MODE	Sets the date display type to be saved or queries the current setting.	4-17
:MISC:DATE:YEAR	Sets the year or queries the current setting.	4-17
:MISC:DATE:MONTH	Sets the month or queries the current setting.	4-17
:MISC:DATE:DAY	Sets the day or queries the current setting.	4-17
:MISC:DATE:HOuR	Sets the hour or queries the current setting.	4-17
:MISC:DATE:MINute	Sets the minute or queries the current setting.	4-17
:MISC:DATE:SECond	Sets the second or queries the current setting.	4-17
:MISC:DATE:SET	Applies the date and time change.	4-17
:MISC:LANGuage	Sets the language or queries the current setting.	4-17
:MISC:RLOsmode	Sets the reflection display or queries the current setting.	4-17
NETWork Group		
:NETWork:CONTRol:PASSword	Sets the Password or queries the current setting.	4-18
:NETWork:CONTRol:TIMEout	Sets the Timeout Value or queries the current setting.	4-18
:NETWork:CONTRol:USERname	Sets the User Name or queries the current setting.	4-18
:NETWork:DHCP	Sets enabling or disabling the DHCP function or queries the current setting.	4-18
:NETWork:GATeway	Sets the gateway or queries the current setting.	4-18
:NETWork:IPAdDress	Sets the IP address or queries the current setting.	4-18
NETWork:NETMask	Sets the netmask or queries the current setting.	4-18
:NETWork:STATe	Sets enabling or disabling the Network or queries the current setting.	4-18
SETup Group		
:SETup:INITialize	Initializes all the settings to factory default.	4-19
STATus Group		
:STATus?	Queries all settings related to the communication status function.	4-19
:STATus:CONDition?	Queries the contents of the condition register.	4-19
:STATus:ERRor?	Queries the error code and message information (top of the error queue).	4-19
:STATus:QENable	Sets whether to store messages other than errors to the error queue or queries the current setting.	4-19
:STATus:QMESsage	Sets whether or not to attach message information to the response to the “:STATus:ERRor?” query or queries the current setting.	4-19
WAVedata Group		
:WAVedata:LENGth?	Queries the number of waveform data values.	4-20
:WAVedata:DISPlay:SEND:ASCIi?	Queries the display waveform data in ASCII format.	4-20
:WAVedata:DISPlay:SEND:BINary?	Queries the display waveform data in binary format.	4-20
:WAVedata:OLDType:DISPlay:SEND?	Queries the display waveform data in Dot 4 format.	4-20
:WAVedata:OLDType:SEND?	Queries the waveform data in Dot 4 format.	4-20
:WAVedata:SEND:ASCIi?	Queries the waveform data in ASCII format.	4-20
:WAVedata:SEND:BINary?	Queries the waveform data in binary format.	4-20
:WAVedata:SEND:START	Sets the start distance of the wavedata or queries the current setting.	4-20
:WAVedata:SEND:SIZE	Sets the number of waveform data to acquire or queries the current setting.	4-20
SYSTem Group		
:SYSTem:REBoot	Execution of restarting	4-21

4.2 Common Commands

***CLS(Clear Status)**

Function Clears all event status registers, the summary of which is reflected in the status byte register.

Syntax *CLS

Example *CLS

Description

- Clears all queues, with the exception of the output queue, and all event registers, with the exception of the MAV summary message.
- After executing this command, OCIS (Operation Complete Command Idle State) and OQIS (Operation Complete Query Idle State) are brought about.

***ESE(Standard Event Status Enable)**

Function Sets/queries the standard event enable register.

Syntax *ESE<wsp><integer>
*ESE?
<integer> = 0 to 255

Example *ESE 251
*ESE? -> 251

Description

- An item having had its bit set becomes enabled.
- Resets to the default value in the following cases:
 - When power is ON
 - When "0" is set
- The set value remains the same in the following cases:
 - *RST
 - *CLS
 - Device clear (DCL, SDC)
- The default is 0.

***ESR?(Standard Event Status Register)**

Function Queries the standard event status register and simultaneously clears it.

Syntax *ESR?

Example *ESR? -> 251

Description The return value of this query is not affected by ESE (Event Status Enable Register).

***IDN? (Identification)**

Function Queries the instrument type and firmware version.

Syntax *IDN?

Example *IDN? -> YOKOGAWA,AQ7277-B01,
SN123456789,F1.00
SN123456789: Serial number
(9 digit string)
F1.00:Firmware version

Description Outputs 4 field data delimited by a comma.

Field 1: Manufacturer "YOKOGAWA"
Field 2: Model "AQ7277-B01"
Field 3: Instrument serial number "SN123456789"
Field 4: Software version "F1.00"

***RST (Reset)**

Function Executes a device reset to return the instrument to the known (default) status.

Syntax *RST

Example *RST

Description

- Stops operation being processed and returns the instrument to the known set value (default value) immediately.
- This unit's parameters are cleared.
- The following items will remain the same. Output queue
SRE
ESE
Calibration data affecting the instrument's specifications

***SRE(Service Request Enable)**

Function Sets/queries the service request enable register.

Syntax *SRE <wsp><integer>
*SRE?
<integer> = 0 to 255

Example *SRE 250
*SRE? -> 250

Description

- An item having had its bit set becomes enabled.
- Resets to the default value in the following cases:
 - When power is ON
 - When "0" is set
- The set value remains the same in the following cases:
 - *RST
 - *CLS
 - Device clear (DCL, SDC)
- The default is 0.

***STB?(Read Status Byte)**

Function Queries the current value of the status byte register.

Syntax *STB?

Example *STB? -> 251

Description STB will not be cleared even when the contents of the register are read.

***TST?(Self Test)**

Function Performs the instrument's self-test and queries the status.

Syntax *TST?

Example *TST? -> 0

Description Executes the operations below among the initialization sequence at startup and outputs the results. The screen retains the waveform screen while the initialization is in progress.
Internal battery check
Internal memory read check
"0" is returned if both checks are successful, "1" if they are not.

4.3 Device-Specific Commands

ACQuire Group

The commands in this group deal with the waveform acquisition. Change the measurement conditions through remote control while the measurement is stopped. You cannot change the measurement conditions while realtime or averaging measurement is in progress. Only :REALtime: STOP or AVERage:STOP is valid while the measurement is in progress.

:ACQuire:ATTenuation

Function Sets the attenuation or queries the current setting.

Syntax :ACQuire:ATTenuation {<Nrf>|AUTO}
:ACQuire:ATTenuation?
<Nrf> = 0.00 to 27.50 (steps of 2.5)

Example :ACQuire:ATTenuation 2.5
:ACQuire:ATTenuation AUTO
:ACQuire:ATTenuation? ->
:ACQ:ATT 25.00

Description The selectable range varies depending on the wavelength, distance range, and pulse width. You cannot set the attenuation if the average mode is high reflection. You cannot set the pulse width if the distance range is set to auto.

:ACQuire:AUTO:ATTenuation?

Function Queries the attenuation for AUTO.

Syntax :ACQuire:AUTO:ATTenuation?
Example :ACQuire:AUTO:ATTenuation? ->
:ACQ:AUT:ATT 0.00

Description The unit is set to m. (meter)

:ACQuire:AUTO:DRANge?

Function Queries the distance range for AUTO.

Syntax :ACQuire:AUTO:DRANge?
Example :ACQuire:AUTO:DRANge? ->
:ACQ:AUT:DRAN 20000

Description The unit is set to s. (second)

:ACQuire:AUTO:PWIDth?

Function Queries the pulse width for AUTO.

Syntax :ACQuire:AUTO:PWIDth?
Example :ACQuire:AUTO:PWIDth? ->
:ACQ:AUT:PWID 100E-09

:ACQuire:AVERage:COUNT?

Function Queries the current average count.

Syntax :ACQuire:AVERage:COUNT?
Example :ACQuire:AVERage:COUNT? ->
:ACQ:AVER:COUN 0

:ACQuire:AVERage:INDEX

Function Sets the average count or queries the current setting.

Syntax :ACQuire:AVERage
:INDEX {AUTO|E2_10|E2_11.....E2_20}
:ACQuire:AVERage:INDEX?
Example :ACQuire:AVERage:INDEX AUTO
:ACQuire:AVERage:INDEX E2_12
:ACQuire:AVERage:INDEX? ->
:ACQ:AVER:IND E2_15

Description Setting the averaging count

Auto:	AUTO
2E+10:	E2_10
2E+11:	E2_11
.....	
2E+20:	E2_20

:ACQuire:AVERage:MODE

Function Sets the average mode or queries the current setting.

Syntax :ACQuire:AVERage
:MODE {HIREFLECTION|HISPEED}
:ACQuire:AVERage:MODE?
Example :ACQuire:AVERage:MODE HIREFLECTION
:ACQuire:AVERage:MODE? ->
:ACQ:AVER:MODE HISPEED

Description High Reflection : HIREFLECTION
High Speed : HISPEED

:ACQuire:AVERage:START

Function Executes the averaging measurement.

Syntax :ACQuire:AVERage:START
Example :ACQuire:AVERage:START

Description This command is valid while the averaging measurement is only in progress.

:ACQuire:AVERage:STOP

Function Stops the averaging measurement.

Syntax :ACQuire:AVERage:STOP

Example :ACQuire:AVERage:STOP

:ACQuire:AVERage:TIME

Function Sets the average time or queries the current setting.

Syntax :ACQuire:AVERage:TIME {<NRf>|AUTO}

:ACQuire:AVERage:TIME?

<NRf> = 2 to 1800

Example :ACQuire:AVERage:TIME AUTO

:ACQuire:AVERage:TIME 1200

:ACQuire:AVERage:TIME? ->

:ACQ:AVER:TIME 60

Description The following value can be set.

Auto: AUTO

2 sec: 2

5 sec: 5

10 sec: 10

20 sec: 20

30 sec: 30

1 min: 60

3 min: 180

5 min: 300

10 min: 600

20 min: 1200

30 min: 1800

:ACQuire:AVERage:TYPE

Function Sets the average unit or queries the current setting.

Syntax :ACQuire:AVERage

:TYPE {TIMES|DURATION}

:ACQuire:AVERage:TYPE?

Example :ACQuire:AVERage:TYPE TIMES

:ACQuire:AVERage:TYPE DURATION

:ACQuire:AVERage:TYPE? ->

:ACQ:AVER:TYPE DURATION

Description Times: TIMES

Duration: DURATION

:ACQuire:DRANge

Function Sets the distance range or queries the current setting.

Syntax :ACQuire:DRANge {<NRf>|AUTO}

:ACQuire:DRANge?

<NRf> = 500m to 400000m

(0.5km to 400km)

Example :ACQuire:DRANge AUTO

:ACQuire:DRANge 500

:ACQuire:DRANge 500m

:ACQuire:DRANge 5000

:ACQuire:DRANge 100km

:ACQuire:DRANge? -> :ACQ:DRAN 1000

Description The value that you can specify varies depending on the wavelength setting.

The selectable distance range varies depending on the wavelength

Setting the distance range

Auto: AUTO

500 m: 500

1 km: 1000

2 km: 2000

5 km: 5000

10 km: 10000

20 km: 20000

50 km: 50000

100 km: 100000

200 km: 200000

300 km: 300000

400 km: 400000

:ACQuire:OFFSet

Function Sets the horizontal measurement start position or queries the current setting.

Syntax :ACQuire:OFFSet {<NRf>}

:ACQuire:OFFSet?

Example :ACQuire:OFFSet 1000

:ACQuire:OFFSet? ->

v:ACQ:OFFS 999.99265E+00

Description The unit is set to m. (meter)

4.3 Device-Specific Commands

:ACquire:PWIDth

Function Sets the pulse width or queries the current setting.

Syntax :ACquire:PWIDth {<NRf>|AUTO}
:ACquire:PWIDth?
<NRf> = 3ns to 20us (3E-9 to 20E-6)

Example :ACquire:PWIDth AUTO
:ACquire:PWIDth 3
:ACquire:PWIDth 3E-9
:ACquire:PWIDth 3ns
:ACquire:PWIDth 10us
:ACquire:PWIDth 10E-6
:ACquire:PWIDth? ->
:ACQ:PWID 20.0E-06

Description The value that you can specify varies depending on the wavelength and distance range. You cannot set the pulse width if the distance range is set to auto.

The value that you can specify varies depending on the wavelength and distance range.

You cannot set the pulse width if the distance range is set to auto.

Auto:	AUTO
3 ns:	3E-09
10 ns:	10E-09
20 ns:	20E-09
50 ns:	50E-09
100 ns:	100E-09
200 ns:	00E-09
500 ns:	500E-09
1 us:	1E-06
2 us:	2E-06
5 us:	5E-06
10 us:	10E-06
20 us:	20E-06

:ACquire:REALtime:START

Function Executes the realtime measurement.

Syntax :ACquire:REALtime:START

Example :ACquire:REALtime:START

Description This command is valid while the measurement is stopped.

:ACquire:REALtime:STOP

Function Stops the realtime measurement.

Syntax :ACquire:REALtime:STOP

Example :ACquire:REALtime:STOP

Description This command is valid while the realtime measurement is only in progress.

:ACquire:SMPinterval:DATA

Function Sets the sampling interval or queries the current setting.

Syntax :ACquire:SMPinterval
:DATA {<NRf>|NORMAL|HI}
:ACquire:SMPinterval:DATA?

Example :ACquire:SMPinterval:DATA 2.0
:ACquire:SMPinterval:DATA NORMAL
:ACquire:SMPinterval:DATA? ->
:ACQ:SMP:DATA HI

Description You only set NORMAL or HI if the distance range is set to auto.

You only set NORMAL or HI if the distance range is set to auto.

Normal:	NORMAL
High resolution:	HI
5 cm:	0.05
10 cm:	0.10
20 cm:	0.20
50 cm:	0.50
1 m:	1.00
2 m:	2.00
4 m:	4.00
8 m:	8.00
16 m:	16.00
32 m:	32.00

:ACquire:SMPinterval:VALue?

Function Queries the sampling interval.

Syntax :ACquire:SMPinterval:VALue?

Example :ACquire:SMPinterval:VALue? ->
:ACQ:SMP:VAL 8.0

:ACquire:WAVelength

Function Sets the measured wavelength or queries the current setting.

Syntax :ACquire:WAVelength {<NRf>}
:ACquire:WAVelength?
<NRf> = 0.850um to 1.650um
(850E-9 to 1650E-9)

Example :ACquire:WAVelength 0.85um
:ACquire:WAVelength 1650E-9
:ACquire:WAVelength 1.650E-6
:ACquire:WAVelength? ->
:ACQ:WAV 1550E-09

Description The wavelength that you can specify varies depending on the model.

:ACQUIRE:SELECTION:DRANGE?

Function Queris the selectable distance range.

Syntax :ACQUIRE:SELECTION:DRANGE?

Sample :ACQUIRE:SELECTION:DRANGE? ->
:ACQUIRE:SELECTION:DRANGE 500,
1000, ... ,4000000

Description The value that you can specify varies depending on the wavelength setting.

The selectable distance range varies depending on the wavelength.

Setting the distance range

Auto: AUTO

500m: 500

1km: 1000

2km: 2000

5km: 5000

10km: 10000

20km: 20000

50km: 50000

100km: 100000

200km: 200000

300km: 300000

400km: 400000

:ACQUIRE:SELECTION:WAVELENGTH?

Function Queris the selectable measured wavelength.

Syntax :ACQUIRE:SELECTION:WAVELENGTH?

Sample :ACQUIRE:SELECTION:WAVELENGTH? ->
:ACQUIRE:SELECTION:WAVELENGTH
1.310E-6 1.1550E-6

:ACQUIRE:SELECTION:PWIDTh?

Function Queris the selectable pulse width.

Syntax :ACQUIRE:SELECTION:PWIDTh?

Sample :ACQUIRE:SELECTION:PWIDTh? ->
:ACQUIRE:SELECTION:PWIDTh
3E-9 1.0E-9, ... , 20E-6

:ACQUIRE:SELECTION:ATTenuation?

Function Queris the selectable attenuatio.

Syntax :ACQUIRE:SELECTION:ATTenuation?

Sample :ACQUIRE:SELECTION:ATTENUATION? ->
:ACQUIRE:SELECTION:ATTENUATION
0.00,2.50,5.00, ... , 27.50

:ACQUIRE:SELECTION:SMPinterval?

Function Queris the selectable sampling interval.

Syntax :ACQUIRE:SELECTION:SMPinterval?

Sample :ACQUIRE:SELECTION:SMPINTERVAL? ->
:ACQUIRE:SELECTION:SMPINTERVAL
0.05,0.10,0.20,...,4.00

ANALYSIS Group

The commands in this group deal with waveform analysis.

:ANALYSIS:ASEarch:EXECute

Function Executes auto detection.

Syntax :ANALYSIS:ASEarch:EXECute

Example :ANALYSIS:ASEarch:EXECute

:ANALYSIS:ASEarch:NUMber?

Function Queries the number of auto detection events.

Syntax :ANALYSIS:ASEarch:NUMber?

Example :ANALYSIS:ASEarch:NUMber? ->
:ANAL:ASE:NUM 2

:ANALYSIS:BCOefficient

Function Sets the backscattering light level of the current wavelength or queries the current setting.

Syntax :ANALYSIS:BCOefficient {<NRf>}
:ANALYSIS:BCOefficient?
<NRf> = -10.00 to -64.99
(steps of 0.01)

Example :ANALYSIS:BCOefficient -25.00
:ANALYSIS:BCOefficient? ->
:ANAL:BCO -50.00

:ANALYSIS:IOR

Function Sets the group refraction index of the current wavelength or queries the current setting.

Syntax :ANALYSIS:IOR {<NRf>}
:ANALYSIS:IOR?
<NRf> = 1.30000 to 1.79999
(steps of 0.00001)

Example :ANALYSIS:IOR 1.48000
:ANALYSIS:IOR? -> :ANAL:GINd 1.48000

:ANALYSIS:CUSor:DELeTe

Function Clears the cursor.

Syntax :ANALYSIS:CUSor:DELeTe

Example :ANALYSIS:CUSor:DELeTe

:ANALYSIS:CUSor:DISTance

Function Sets the cursor position or queries the current setting.

Syntax :ANALYSIS:CUSor:DISTance {<NRf>}
:ANALYSIS:CUSor:DISTance?
<NRf> = Depend on distance range,
distance reference and IOR.

Example :ANALYSIS:CUSor:DISTance 10.00
:ANALYSIS:CUSor:DISTance? ->
:ANAL:CUS:DIST 11.529900E+00

Description Depending on the distance range, distance reference and group refraction.

4.3 Device-Specific Commands

:ANALysis:CURSor:DECibel?

Function Queries the cursor dB (decibel).

Syntax :ANALysis:CURSor:DECibel?

Example :ANALysis:CURSor:DECibel? ->
:ANAL:CURS:DEC 32.878

:ANALysis:CURSor:LINK

Function Sets the cursor link or queries the cursor link setting.

Syntax :ANALysis:CURSor:LINK {<Boolean>}
:ANALysis:CURSor:LINK?

Sample :ANALYSIS:CURSOR:LINK ON
:ANALYSIS:CURSOR:LINK? ->
:ANALYSIS:CURSOR:LINK 1

:ANALysis:EMARker:LMTechnique

Function Sets the approximation method (event) or queries the current setting.

Syntax :ANALysis:EMARker
:LMTechnique {LSA|TPA}
:ANALysis:EMARker:LMTechnique?

Example :ANALysis:EMARker:LMTechnique LSA
:ANALysis:EMARker:LMTechnique? ->
:ANAL:EMAR:LMT TPA

:ANALysis:EMARker:SET:M1

Function Sets marker M1 of the current event or queries the current setting.

Syntax :ANALysis:EMARker:SET:M1
:ANALysis:EMARker:SET:M1?

Example :ANALysis:EMARker:SET:M1
:ANALysis:EMARker:SET:M1? ->
:ANAL:EMAR:SET:M1 5.0471900E+03

Description The marker is set to the cursor position when you set the marker.

:ANALysis:EMARker:SET:M2

Function Sets marker M2 of the current event or queries the current setting.

Syntax :ANALysis:EMARker:SET:M2
:ANALysis:EMARker:SET:M2?

Example :ANALysis:EMARker:SET:M2
:ANALysis:EMARker:SET:M2? ->
:ANAL:EMAR:SET:M2 7.2463500E+03

Description The marker is set to the cursor position when you set the marker.

:ANALysis:EMARker:SET:M3

Function Sets marker M3 of the current event or queries the current setting.

Syntax :ANALysis:EMARker:SET:M3
:ANALysis:EMARker:SET:M3?

Example :ANALysis:EMARker:SET:M3
:ANALysis:EMARker:SET:M3? ->
:ANAL:EMAR:SET:M3 9.0471600E+03

Description The marker is set to the cursor position when you set the marker.

:ANALysis:EMARker:SET:Y2

Function Sets marker Y2 of the current event or queries the current setting.

Syntax :ANALysis:EMARker:SET:Y2
:ANALysis:EMARker:SET:Y2?

Example :ANALysis:EMARker:SET:Y2
:ANALysis:EMARker:SET:Y2? ->
:ANAL:EMAR:SET:Y2 7.3777700E+03

Description The marker is set to the cursor position when you set the marker.

:ANALysis:DUNit

Function Sets the distance unit or queries the current setting.

Syntax :ANALysis:DUNit {KM|MILE|KF}
:ANALysis:DUNit?

Example :ANALysis:DUNit KM
:ANALysis:DUNit? -> :ANAL:DUN MILE

Description Setting the Distance Unit

km: KM
mile: MILE
kf: KF

:ANALysis:EVENT:CURRENT:INDEX

Function Changes the current event.

Syntax :ANALysis:EVENT:CURRENT
:INDEX {<Nrf>}
<Nrf> = Event number

Example :ANALysis:EVENT:CURRENT:INDEX 5

Description The event of the specified number is the current event.

Event number: 1 to 100
S point, R point: 1
E point: 0

:ANALysis:EVENT:CURRENT:NOTE

Function Sets the event note or queries the current setting.

Syntax :ANALysis:EVENT:CURRENT
:NOTE {<Character string>}
:ANALysis:EVENT:CURRENT:NOTE?
<Character string> = up to 36 characters

Example :ANALysis:EVENT:CURRENT:NOTE "aaa"
:ANALysis:EVENT:CURRENT:NOTE? ->
:ANAL:EVEN:CURR:NOTE "123"

Description The index number must be set ahead of time with the :ANALysis:EVENT:CURRENT:INDEX command before using this command.
Enter a comment for the current event.
You can enter the comment using up to 36 characters.

:ANALysis:EVENT:IOR

Function Sets the section IOR of the current event or queries the current setting.

Syntax :ANALysis:EVENT:IOR {<NRf>}
 :ANALysis:EVENT:IOR?
 <NRf> = 1.30000 to 1.79999
 (steps of 0.00001)

Example :ANALysis:EVENT:IOR 1.47:ANALysis
 :EVENT:IOR? ->
 :ANAL:EVENT:GIND 1.46000

:ANALysis:EVENT:CURRENT:PEAK?

Function Retrieves the peak decibel of the current event.

Syntax :ANALysis:EVENT:CURRENT:LOSS?
 Sample :ANALYSIS:EVENT:CURRENT:LOSS? ->
 2.000

Description The index number must be set ahead of time with the :ANALysis:EVENT:CURRENT:INDEX command before using this command.

:ANALysis:EVENT:FIX

Function Event Fix

Syntax :ANALysis:EVENT:FIX <Boolean>
 :ANALysis:EVENT:FIX?

Sample :ANALYSIS:EVENT:FIX ON
 :ANALYSIS:EVENT:FIX? ->
 :ANALYSIS:EVENT:FIX 1

:ANALysis:EVENT:CURRENT:DISTance?

Function Retrieves the distance of the current event.

Syntax :ANALysis:EVENT:CURRENT:DISTance?
 Example :ANALYSIS:EVENT:CURRENT:DISTANCE? ->
 987.000

Description The index number must be set ahead of time with the :ANALysis:EVENT:CURRENT:INDEX command before using this command.

:ANALysis:EVENT:CURRENT:LOSS?

Function Retrieves the splice loss of the current event.

Syntax :ANALysis:EVENT:CURRENT:LOSS?
 Example :ANALYSIS:EVENT:CURRENT:LOSS? ->
 2.000

Description The index number must be set ahead of time with the :ANALysis:EVENT:CURRENT:INDEX command before using this command.

:ANALysis:EVENT:CURRENT:RETurnloss?

Function Retrieves the return loss of the current event.

Syntax :ANALysis:EVENT:CURRENT:RETurnloss?
 Example :ANALYSIS:EVENT:CURRENT
 :RETURNLOSS? -> 2.000

Description The index number must be set ahead of time with the :ANALysis:EVENT:CURRENT:INDEX command before using this command.

:ANALysis:EVENT:CURRENT:CUMLoss?

Function Retrieves the accumulated loss of the current event.

Syntax :ANALysis:EVENT:CURRENT:CUMLoss?
 Example :ANALYSIS:EVENT:CURRENT:CUMLOSS? ->
 1.810

Description The index number must be set ahead of time with the :ANALysis:EVENT:CURRENT:INDEX command before using this command.

:ANALysis:EVENT:CURRENT:UNITloss?

Function Retrieves the loss per unit (dB/m) of the current event.

Syntax :ANALysis:EVENT:CURRENT:UNITloss?
 Example :ANALYSIS:EVENT:CURRENT:UNITLOSS? ->
 2.000

Description The index number must be set ahead of time with the :ANALysis:EVENT:CURRENT:INDEX command before using this command.

:ANALysis:EVENT:CURRENT:TYPE?

Function Retrieves the event type of the current event.

Syntax :ANALysis:EVENT:CURRENT:TYPE?
 Example :ANALYSIS:EVENT:CURRENT:TYPE? ->
 REFLECTANCE

Description The index number must be set ahead of time with the :ANALysis:EVENT:CURRENT:INDEX command before using this command.

Reflection: REFLECTance
 Positive loss: SPLus
 Negative loss: SMINus

:ANALysis:EVENT:CURRENT:IOR?

Function Retrieves the section IOR of the current event.

Syntax :ANALysis:EVENT:CURRENT:IOR?
 Example :ANALYSIS:EVENT:CURRENT:IOR? ->
 1.46000

Description The index number must be set ahead of time with the :ANALysis:EVENT:CURRENT:INDEX command before using this command.

:ANALysis:EVENT:DELeTe

Function Deletes the current event.

Syntax :ANALysis:EVENT:DELeTe
 Example :ANALysis:EVENT:DELeTe

:ANALysis:EVENT:INSert

Function Inserts the event at the cursor position.

Syntax :ANALysis:EVENT:INSert
 Example :ANALysis:EVENT:INSert

:ANALysis:FEDetection

Function Sets the fault event display or queries the current setting.

Syntax :ANALysis:FEDetection {<Boolean>}
 :ANALysis:FEDetection?
 Example :ANALysis:FEDetection ON
 :ANALysis:FEDetection? -> :ANAL:FED 0

4.3 Device-Specific Commands

:ANALysis:FMARker:DELeTe

Function Deletes the marker.

Syntax :ANALysis:FMARker:DELeTe

Example :ANALysis:FMARker:DELeTe

:ANALysis:FMARker:LMTechnique

Function Sets the approximation method of the marker or queries the current setting.

Syntax :ANALysis:FMARker
:LMTechnique {LSA|TPA}
:ANALysis:FMARker:LMTechnique?

Example :ANALysis:FMARker:LMTechnique LSA
:ANALysis:FMARker:LMTechnique? ->
:ANAL:FMAR:LMT TPA

Description Least squares approximation: LSA
Two point approximation: TPA

:ANALysi s:FMARker:LOSS?

Function Queries the splice loss.

Syntax :ANALysis:FMARker:LOSS?

Example :ANALysis:FMARker:LOSS? ->
:ANAL:FMAR:LOSS 162.00000E-03

:ANALysis:FMARker:RETurnloss:VALue?

Function Queries the return loss.

Syntax :ANALysis:FMARker:RETurnloss:VALue?

Example :ANALysis:FMARker:RETurnloss
:VALue? -> :ANAL:FMAR:RET:VAL
47.003000E+00

:ANALysis:FMARker:RETurnloss: SATurated?

Function Queries the saturation of the return loss.

Syntax :ANALysis:FMARker:RETurnloss
:SATurated?

Example :ANALysis:FMARker:RETurnloss
:SATurated? -> :ANAL:FMAR:RET:SAT 1

Description Unsaturated: 0
Saturated: 1

:ANALysis:FMARker:REFLection:VALue?

Function Queries the reflection level.

Syntax :ANALysis:FMARker:REFLection:VALue?

Example :ANALysis:FMARker:REFLection
:VALue? -> :ANAL:FMAR:REFL:VAL
1.0640000E+00

:ANALysis:FMARker:REFLection: SATurated?

Function Queries the saturation of the reflection level.

Syntax :ANALysis:FMARker:REFLection
:SATurated?

Example :ANALysis:FMARker:REFLection
:SATurated? -> :ANAL:FMAR:REFL:SAT 0

Description Unsaturated: 0
Saturated: 1

:ANALysis:FMARker:LEFT:LOSS?

Function Queries the loss between markers 1 and 2.

Syntax :ANALysis:FMARker:LEFT:LOSS?

Example :ANALysis:FMARker:LEFT:LOSS? ->
:ANAL:FMAR:LEFT
:LOSS 137.00000E-03

:ANALysis:FMARker:LEFT:DIStance?

Function Queries the distance between markers 1 and 2.

Syntax :ANALysis:FMARker:LEFT:DIStance?

Example :ANALysis:FMARker:LEFT:DIStance? ->
:ANAL:FMAR:LEFT
:DIST 490.20000E+00

:ANALysis:FMARker:LEFT:UNITloss?

Function Queries the slope between markers 1 and 2.

Syntax :ANALysis:FMARker:LEFT:UNITloss?

Example :ANALysis:FMARker:LEFT:UNITloss? ->
:ANAL:FMAR:LEFT
:UNIT 279.00000E-03

:ANALysis:FMARker:RIGHT:LOSS?

Function Queries the loss between markers 2 and 3.

Syntax :ANALysis:FMARker:RIGHT:LOSS?

Example :ANALysis:FMARker:RIGHT:LOSS? ->
:ANAL:FMAR:RIGH
:LOSS -159.00000E-03

:ANALysis:FMARker:RIGHT:DIStance?

Function Queries the distance between markers 2 and 3.

Syntax :ANALysis:FMARker:RIGHT:DIStance?

Example :ANALysis:FMARker:RIGHT
:DIStance? -> :ANAL:FMAR:RIGH
:DIST 232.95000E+00

:ANALysis:FMARker:RIGHT:UNITloss?

Function Queries the slope between markers 2 and 3.

Syntax :ANALysis:FMARker:RIGHT:UNITloss?

Example :ANALysis:FMARker:RIGHT
:UNITloss? -> :ANAL:FMAR:RIGH
:UNIT -683.00000E-03

:ANALysis:FMARker:SET:M<x>

Function Sets the marker or queries the current setting.

Syntax :ANALysis:FMARker:SET:M<x>
:ANALysis:FMARker:SET:M<x>?
<x> = 1,2,3

Example :ANALysis:FMARker:SET:M1
:ANALysis:FMARker:SET:M1? -> ANAL
:FMAR:SET:M1 7.2648300E+03

:ANALysis:FMARker:SET:Y<x>

Function Sets the auxiliary marker or queries the current setting.

Syntax :ANALysis:FMARker:SET:Y<x>
:ANALysis:FMARker:SET:Y<x>?
<x> = 1,2,3

Example :ANALysis:FMARker:SET:Y2
:ANALysis:FMARker:SET:Y2? -> ANAL
:FMAR:SET:Y2 6.1786000E+03

:ANALysis:REFeRence:DELeTe

Function Deletes the distance reference.

Syntax :ANALysis:REFeRence:DELeTe

Example :ANALysis:REFeRence:DELeTe

:ANALysis:REFeRence:DISTance

Function Sets the distance reference or queries the current setting.

Syntax :ANALysis:REFeRence:DISTance

:ANALysis:REFeRence:DISTance?

Example :ANALysis:REFeRence:DISTance

:ANALysis:REFeRence:DISTance? -> ANAL
:REF:DIST 39.999710E+03

Description The query's response is in 'm' units based on the near end of the OTDR.

With a dummy fiber, the OTDR near end is approximately 100 m.

:ANALysis:THReShold:EOFiBer

Function Sets the threshold level of the end of fiber or queries the current setting.

Syntax :ANALysis:THReShold:EOFiBer {<NRf>}

:ANALysis:THReShold:EOFiBer?

<NRf> = 3 to 10

Example :ANALysis:THReShold:EOFiBer 5

:ANALysis:THReShold:EOFiBer? ->

:ANAL:THR:EOF 10

:ANALysis:THReShold:RLOSS

Function Sets the threshold level of the return loss or queries the current setting.

Syntax :ANALysis:THReShold:RLOSS {<NRf>}

:ANALysis:THReShold:RLOSS?

<NRf> = 20 to 70

Example :ANALysis:THReShold:RLOSS 50

:ANALysis:THReShold:RLOSS? ->

:ANAL:THR:RLOS 30

:ANALysis:THReShold:SLOSS

Function Sets the threshold level of the splice loss or queries the current setting.

Syntax :ANALysis:THReShold:SLOSS {<NRf>}

:ANALysis:THReShold:SLOSS?

<NRf> = 0.01 to 9.99

Example :ANALysis:THReShold:SLOSS 0.01

:ANALysis:THReShold:SLOSS? ->

:ANAL:THR:SLOS 1.23

:ANALysis:TRACefix:STATe

Function Sets the tracefix or queries the current setting.

Syntax :ANALysis:TRACefix:

STATe {<Boolean>}

Example :ANALysis:STRACefix:STATe ON

:ANALysis:STRACefix:STATe? -> }

:ANALysis:STRACefix:STATe 1

COMMunicate Group

The commands in this group deal with communications.

:COMMunicate?

Function Queries all settings related to communications.

Syntax :COMMunicate?

Example :COMMUNICATE? -> :COM:HEAD 1;VERB 0

:COMMunicate:HEADer

Function Sets whether to include a header in the response to a query or queries the current setting.

Syntax :COMMunicate:HEADer {<Boolean>}

:COMMunicate:HEADer?

Example :COMMUNICATE:HEADER ON

:COMMUNICATE:HEADER? ->:COMM:HEAD 1

:COMMunicate:VERBoSe

Function Sets whether to return the response in full or abbreviated form or queries the current setting.

Syntax :COMMunicate:VERBoSe {<Boolean>}

:COMMunicate:VERBoSe?

Example :COMMUNICATE:VERBOSE ON

:COMMUNICATE:VERBOSE?

->:COMM:VERB 0

4.3 Device-Specific Commands

DISPlay Group

The commands in this group deal with the screen display.

:DISPlay:ALINe

Function Sets the display of the approximation line or queries the current setting.

Syntax :DISPlay:ALINe {<Boolean>}
:DISPlay:ALINe?

Example :DISPlay:ALINe ON
:DISPlay:ALINe? -> :DISP:ALIN 0

:DISPlay:COLor

Function Sets the screen color or queries the current setting.

Syntax :DISPlay:COLor {COLOR1|COLOR2|COLOR3|BLACKWHITE}:DISPlay:COLor?

Example :DISPlay:COLor COLOR1
:DISPlay:COLor? -> :DISP:COL COLOR2

Description Setting the screen color

Color 1: COLOR1
Color 2: COLOR2
Color 3: COLOR3
B & W: BLACKWHITE

:DISPlay:CURSor:DBValue

Function Sets the dB value of cursor or queries the current setting..

Syntax :DISPlay:CURSor:DBValue {<Boolean>}
:DISPlay:CURSor:DBValue?

Example :DISPlay:CURSor:DBValue ON
:DISPlay:CURSor:DBValue? ->
:DISPlay:CURSor:DBValue 1

:DISPlay:CURSor:SECond

Function Sets the secondary cursor display or queries the current setting.

Syntax :DISPlay:CURSor:SECond {<Boolean>}
:DISPlay:CURSor:SECond?

Example :DISPlay:CURSor:SECond ON
:DISPlay:CURSor:SECond? ->
:DISP:CURS:SEC 1

:DISPlay:CURSor:TYPE

Function Sets the cursor type or queries the current setting.

Syntax :DISPlay:CURSor:TYPE {CROSS|LINE}
:DISPlay:CURSor:TYPE?

Example :DISPlay:CURSor:TYPE LINE
:DISPlay:CURSor:TYPE? ->
:DISP:CURS:TYPE CROSS

Description Cross (+): CROSS
Line (!): LINE

:DISPlay:DECibel:UPPer

Function Sets the display start level or queries the current setting.

Syntax :DISPlay:DECibel:UPPer {<NRF>}
:DISPlay:DECibel:UPPer?
<NRF> = 1.6 to 70
(8 times of dB/Div scale to 70)

Example :DISPlay:DECibel:UPPer 60
:DISPlay:DECibel:UPPer? ->
:DISP:DEC:UPP 50.0

Description The range that you can specify varies depending on the vertical zoom rate.

:DISPlay:DIGit:DECibel

Function Sets dB display digit or queries the current setting.

Syntax :DISPlay:DIGit
:DECibel {DIGIT1|DIGIT2|DIGIT3}
:DISPlay:DIGit:DECibel?

Example :DISPlay:DIGit:DECibel DIGIT3
:DISPlay:DIGit:DECibel? ->
:DISP:DIG:DEC DIGIT1

Description **: *: DIGIT1
*: **: DIGIT2
*: **: DIGIT3

:DISPlay:DIGit:DISTance

Function Sets the distance display unit or queries the current setting.

Syntax :DISPlay:DIGit:DISTance
{DIGIT3|DIGIT4|DIGIT5}
:DISPlay:DIGit:DISTance?

Example :DISPlay:DIGit:DISTance DIGIT5
:DISPlay:DIGit:DISTance? ->
:DISP:DIG:DIST DIGIT4

Description **: **: DIGIT3
Displays the value with four decimal digits.
*: **: DIGIT4
*: **: DIGIT5

:DISPlay:DISTance:LEFT

Function Sets the display start distance or queries the current setting.

Syntax :DISPlay:DISTance:LEFT {<NRF>}
:DISPlay:DISTance:LEFT?

Example :DISPlay:DISTance:LEFT 1000
:DISPlay:DISTance:LEFT? ->
:DISP:DIST:LEFT 498.97000E+00

Description The range that you can specify varies depending on the horizontal zoom rate.

:DISPlay:DIVide:DECibel

Function Sets the vertical zoom rate or queries the current setting.

Syntax :DISPlay:DIVide:DECibel {<NRf>}
 :DISPlay:DIVide:DECibel?
 <NRf> = 0.2 to 7.5
 (0.2, 0.5, 1.0, 2.0, 5.0, 7.5)

Example :DISPlay:DIVide:DECibel 5.0
 :DISPlay:DIVide:DECibel? ->
 :DISP:DIV:DEC 1.0

:DISPlay:DIVide:DISTance

Function Sets the horizontal zoom rate or queries the current setting.

Syntax :DISPlay:DIVide:DISTance {<NRf>}
 :DISPlay:DIVide:DISTance?
 <NRf> = 1m to 40km

Example :DISPlay:DIVide:DISTance 2000
 :DISPlay:DIVide:DISTance? ->
 :DISP:DIV:DIST 10.0E+03

Description The range that you can specify varies depending on the distance range.

400 k range:	50 m to 40 km
300 k range:	50 m to 30 km
200 k range:	50 m to 20 km
100 k range:	50 m to 10 km
50 k range:	20 m to 5 km
20 k range:	10 m to 2 km
10 k range:	5 m to 1 km
5 k range:	5 m to 500 m
2 k range:	2 m to 200 m
1 k range:	1 m to 100 m
500 m range:	1 m to 50 km

:DISPlay:GTYPe

Function Sets grid display or queries the current setting.

Syntax :DISPlay:GTYPe {OFF|LINE|DOTS}
 :DISPlay:GTYPe?

Example :DISPlay:GTYPe OFF
 :DISPlay:GTYPe? -> :DISP:GTYP 0

Description Disable: OFF
 Line: LINE
 Dots: DOTS

:DISPlay:IMARk

Function Sets the marker information display or queries the current setting.

Syntax :DISPlay:IMARk {<Boolean>}
 :DISPlay:IMARk?

Example :DISPlay:IMARk ON
 :DISPlay:IMARk? -> :DISP:IMAR 1

:DISPlay:IScale

Function Initializes the display scale.

Syntax :DISPlay:IScale
 Example :DISPlay:IScale

:DISPlay:OVERview

Function Sets overview display or queries the current setting.

Syntax :DISPlay:OVERview {<Boolean>}
 :DISPlay:OVERview?

Example :DISPlay:OVERview ON
 :DISPlay:OVERview? -> :DISP:OVER 1

:DISPlay:WAVE:TYPE

Function Sets the waveform type display format or queries the current setting.

Syntax :DISPlay:WAVE:TYPE {LINE|DOT}
 :DISPlay:WAVE:TYPE?

Example :DISPlay:WAVE:TYPE LINE
 :DISPlay:WAVE:TYPE? ->
 :DISP:WAV:TYPE DOT

Description You can select the waveform display type from the following.

Connects the sampling data with lines for the display: LINE
 Displays the sampled data with dots: DOT

4.3 Device-Specific Commands

FILE Group

The commands in this group can be used to save event lists and waveform data to files in CSV and other formats.

:FILE:CSV:EVENT:VALId

Function Confirms if it is possible to make the csv format of the event list.

Syntax :FILE:CSV:EVENT:VALId?

Sample :FILE:CSV:EVENT:VALID ON

:FILE:CSV:EVENT:GET

Function Gets the CSV format of the event list.

Syntax :FILE:CSV:EVENT:GET?

Sample :FILE:CSV:EVENT:GET #812345678....

Description Block data format.

:FILE:CSV:TRACe:VALId

Function Confirms if it is possible to make the csv format of the trace data.

Syntax :FILE:CSV:TRACe:VALId?

Sample :FILE:CSV:TRACE:VALID ON

:FILE:CSV:TRACe:GET

Function Gets the CSV format of the trace data.

Syntax :FILE:CSV:TRACe:GET?

Sample :FILE:CSV:TRACE:GET #812345678....

Description Block data format.

:FILE:SOR:VALId?

Function Confirms if it is possible to make the SOR format of the trace data.

Syntax :FILE:SOR:VALId <Boolean>

Sample :FILE:SOR:VALId? -> :FILE:SOR:VALID 0

:FILE:SOR:GET?

Function Gets the sor format of trace.

Syntax :FILE:SOR:GET?

Sample :FILE:SOR:VALId? ->

:FILE:SOR:VALID

#6123456ABCDEFGHJ1234567890

Description Block data format.

MENU Group

The commands in this group are used to set the function or marker mode or query the settings.

:MENU:ERRor:CLEar

Function Deleting the error dialog display

Syntax :MENU:ERRor:CLEar

Example :MENU:ERROR:CLEAR

:MENU:FUNCTion

Function Sets the function mode or queries the current setting.

Syntax :MENU:FUNCTion {OTDR|TOP}

:MENU:FUNCTion?

Example :MENU:FUNCTion OTDR

:MENU:FUNCTion? -> :MENU:FUNC OTDR

Description Top menu: TOP

OTDR: OTDR

:MENU:MARKer

Function Sets the marker mode or queries the current setting.

Syntax :MENU:MARKer {TRACE|LINE}

:MENU:MARKer?

Example :MENU:MARKer TRACE

:MENU:MARKer? -> :MEN:MARK LINE

MISC Group

The commands in this group deal with the date, time, language, and power management.

:MISC:DATE:MODE

Function Sets the date display type to be saved or queries the current setting.

Syntax :MISC:DATE:MODE {TYPE1|TYPE2}
:MISC:DATE:MODE?

Example :MISC:DATE:MODE TYPE1
:MISC:DATE:MODE? ->
:MISC:DATE:MODE TYPE2

Description Setting the display of date
2006/08/29 12:16: TYPE1
08/29/2006 12:16: TYPE2

:MISC:DATE:YEAR

Function Sets the year or queries the current setting.

Syntax :MISC:DATE:YEAR {<Nrf>}
:MISC:DATE:YEAR?
<Nrf> = 2006 to 2036

Example :MISC:DATE:YEAR 2006
:MISC:DATE:YEAR? ->
:MISC:DATE:YEAR 2006

Description This setting will come into effect when MISC:DATE:SET is executed.

:MISC:DATE:MONTH

Function Sets the month or queries the current setting.

Syntax :MISC:DATE:MONTH {<Nrf>}
:MISC:DATE:MONTH?
<Nrf> = 1 to 12

Example :MISC:DATE:MONTH 8
:MISC:DATE:MONTH? -> MISC:DATE:MONT 8

Description This setting will come into effect when MISC:DATE:SET is executed.

:MISC:DATE:DAY

Function Sets the day or queries the current setting.

Syntax :MISC:DATE:DAY {<Nrf>}
:MISC:DATE:DAY?
<Nrf> = 1 to 31

Example :MISC:DATE:DAY 29
:MISC:DATE:DAY? -> :MISC:DATE:DAY 19

Description This setting will come into effect when MISC:DATE:SET is executed.

:MISC:DATE:HOURL

Function Sets the hour or queries the current setting.

Syntax :MISC:DATE:HOURL {<Nrf>}
:MISC:DATE:HOURL?
<Nrf> = 0 to 23

Example :MISC:DATE:HOURL 17
:MISC:DATE:HOURL? ->
:MISC:DATE:HOURL 12

Description This setting will come into effect when MISC:DATE:SET is executed.

:MISC:DATE:MINute

Function Sets the minute or queries the current setting.

Syntax :MISC:DATE:MINute {<Nrf>}
:MISC:DATE:MINute?
<Nrf> = 0 to 59

Example :MISC:DATE:MINute 5
:MISC:DATE:MINute? ->
:MISC:DATE:MIN 59

Description This setting will come into effect when MISC:DATE:SET is executed.

:MISC:DATE:SECond

Function Sets the second or queries the current setting.

Syntax :MISC:DATE:SECond {<Nrf>}
:MISC:DATE:SECond?

Sample :MISC:DATE:SECond 8
:MISC:DATE:SECond? ->
:MISC:DATE:SECond 8

Description This setting will come into effect when MISC:DATE:SET is executed.

:MISC:DATE:SET

Function Applies the date and time change.

Syntax :MISC:DATE:SET
Example :MISC:DATE:SET

:MISC:LANGuage

Function Sets the language or queries the current setting.

Syntax :MISC:LANGuage {ENGLISH|FRENCH|GERMANY|ITALY|SPANISH|TCHINESE|DUTCH|FINNISH|NORWEGIAN|POLISH|CZECH|SWEDISH|JAPANESE|CHINESE|KOREAN|RUSSIAN } :MISC:LANGuage?

Example :MISC:LANGuage ENGLISH
:MISC:LANGuage? ->
:MISC:LANG ENGLISH

:MISC:RLOSSmode

Function Sets the reflection display or queries the current setting.

Syntax :MISC:RLOSSmode {NORMAL | NTT}
:MISC:RLOSSmode?

Sample :MISC:RLOSSMODE NORMAL
:MISC:RLOSSMODE? ->
:MISC:RLOSSMODE NTT

Description Return loss: NORMAL Reflection: NTT

4.3 Device-Specific Commands

NETWork Group

The commands in this group can be used to set and query network settings such as the IP address and user name.

:NETWork:CONTRol:PASSword

Function Sets the Password or queries the current setting.

Syntax :NETWork:CONTRol:PASSword
<character string>

Example :NETWORK:CONTROL:PASSWORD "ABC"
:NETWORK:CONTROL:PASSWORD? ->
:NETWORK:CONTROL:PASSWORD "ABC"

Description It cannot be set if NETWork:STATe is turned OFF.
This command is enabled after restarting. You can restart by executing :SYSTem:REBoot.
A password is not required if the user name is "anonymous."

:NETWork:CONTRol:TIMEout

Function Sets the Timeout Value or queries the current setting.

Syntax :NETWork:CONTRol:TIMEout
{<NRf>|INFinite}:NETWork:CONTRol:TIMEout?<NRf>:1~7200

Example :NETWORK:CONTROL:TIMEOUT 30
:NETWORK:CONTROL:TIMEOUT? ->
:NETWORK:CONTROL:TIMEOUT 30

Description It cannot be set if NETWork:STATe is turned OFF.
This command is enabled in next connecting.

:NETWork:CONTRol:USERname

Function Sets the User Name or queries the current setting.

Syntax :NETWork:CONTRol:USERname
<character string>

Example :NETWORK:CONTROL:USERNAME
"anonymous"
:NETWORK:CONTROL:USERNAME? ->
:NETWORK:CONTROL:USERNAME
"anonymuous"

Description It cannot be set if NETWork:STATe is turned OFF.
This command is enabled after restarting. You can restart by executing :SYSTem:REBoot.

:NETWork:DHCP

Function Sets enabling or disabling the DHCP function or queries the current setting.

Syntax :NETWork:DHCP <Boolean>
:NETWork:DHCP?

Example :NETWORK:DHCP ON
:NETWORK:DHCP? -> :NETWORK:DHCP 1

Description Invalid: OFF or 0
Valid: ON or 1
It cannot be set if NETWork:STATe is turned OFF.
This command is enabled after restarting. You can restart by executing :SYSTem:REBoot.

:NETWork:GATeway

Function Sets the gateway or queries the current setting.

Syntax :NETWork:GATeway <
:NETWork:GATeway?

Example :NETWORK:GATEWAY "255.255.255.0"
:NETWORK:GATEWAY? ->
:NETWORK:GATEWAY "255.255.255.0"

Description It cannot be set if NETWork:STATe is turned OFF.
It cannot be set if NETWork:DHCP is turned OFF.
This command is enabled after restarting. You can restart by executing :SYSTem:REBoot.

:NETWork:IPAdDress

Function Sets the IP address or queries the current setting.

Syntax :NETWork:IPAdDress <
:NETWork:IPAdDress?

Example :NETWORK:IPADDRESS "192.168.0.1"
:NETWORK:IPADDRESS? ->
:NETWORK:IPADDRESS "192.168.0.1"

Description It cannot be set if NETWork:STATe is turned OFF.
It cannot be set if NETWork:DHCP is turned OFF.
This command is enabled after restarting. You can restart by executing :SYSTem:REBoot.

NETWork:NETMask

Function Sets the netmask or queries the current setting.

Syntax :NETWork:NETMask <
:NETWork:NETMask?

Example :NETWORK:NETMASK "255.255.255.0"
:NETWORK:NETMASK? ->
:NETWORK:NETMASK "255.255.255.0"

Description It cannot be set if NETWork:STATe is turned OFF.
It cannot be set if NETWork:DHCP is turned OFF.
This command is enabled after restarting. You can restart by executing :SYSTem:REBoot.

:NETWork:STATe

Function Sets enabling or disabling the Network or queries the current setting.

Syntax :NETWork:STATe <Boolean>
:NETWork:STATe?

Example :NETWORK:STATE ON
:NETWORK:STATE? -> :NETWORK:STATE 1

Description Invalid: OFF or 0
Valid: ON or 1
This command is enabled after restarting. You can restart by executing :SYSTem:REBoot.

SETup Group

The commands in this group deal with the initialization of the settings. You can reset the settings to factory default.

:SETup:INITialize

Function Initializes all the settings to factory default.

STATus Group

The commands in the STATus group are used to make settings and inquiries related to the communication status function.

:STATus?

Function Queries all settings related to the communication status function.

Syntax :STATus?

Example :STATUS? -> :STAT:QEN 1;QMES 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 section 5.4, "Condition Register."

: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: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? -> :STAT:QEN 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? -> :STAT:QMES 1

4.3 Device-Specific Commands

WAVedata Group

The commands in this group deal with the waveform data.

:WAVedata:LENGTH?

Function Queries the number of waveform data values.

Syntax :WAVedata:LENGTH?

Example :WAVedata:LENGTH? -> :WAV:LENG 25000

:WAVedata:DISPlay:SEND:AScii?

Function Queries the display waveform data in ASCII format.

Syntax :WAVedata:DISPlay:SEND:AScii?

Example :WAVedata:DISPlay:SEND:AScii? ->
0.000,1.234,...

:WAVedata:DISPlay:SEND:BINary?

Function Queries the display waveform data in binary format.

Syntax :WAVedata:DISPlay:SEND:BINary?

Example :WAVedata:DISPlay:SEND:BINary? ->
#800000010ABCD...

Description Block data format.

:WAVedata:OLDType:DISPlay:SEND?

Function Queries the display waveform data in Dot 4 format.

Syntax :WAVedata:OLDType:DISPlay:SEND?

Example :WAVedata:OLDType:DISPlay:SEND? ->
#42000...

Description Block data format.

:WAVedata:OLDType:SEND?

Function Queries the waveform data in Dot 4 format.

Syntax :WAVedata:OLDType:SEND?

Example :WAVedata:OLDType:SEND? -> #51000...

Description Block data format.

:WAVedata:SEND:AScii?

Function Queries the waveform data in ASCII format.

Syntax :WAVedata:SEND:AScii?

Example :WAVedata:SEND:AScii? ->
0.000,1.234,...

:WAVedata:SEND:BINary?

Function Queries the waveform data in binary format.

Syntax :WAVedata:SEND:BINary?

Example :WAVedata:SEND:BINary? ->
#6123456ABCD...

Description Block data format.

:WAVedata:SEND:START

Function Sets the start distance of the wavedata or queries the current setting.

Syntax :WAVedata:SEND:START {<Nrf>}
:WAVedata:SEND:START?
<Nrf> = 0 to the measurement
distance (m)

Example :WAVEDATA:SEND:START 20000
:WAVEDATA:SEND:START? ->
:WAVEDATA:SEND:START 20000

Description Numbers exceeding the setting range are rounded.

:WAVedata:SEND:SIZE

Function Sets the number of waveform data to acquire or queries the current setting.

Syntax :WAVedata:SEND:END {<Nrf>}
:WAVedata:SEND:END?
<Nrf> = 1 to the number of
measurement

Example :WAVEDATA:SEND:END {<Nrf>}
:WAVEDATA:SEND:END?
-> :WAVEDATA:SEND:END 20000
<Nrf> = 1 to the number of
measurement

Description Numbers exceeding the setting range are rounded.

SYSTem Group
:SYSTem:REBoot
Function Execution of restarting
Syntax :SYSTem:REBoot
Example :SYSTEM:REBOOT

4.4 Condition Register

The condition register indicates the internal condition of the instrument.

Condition Register	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
:STATus:CONDition?	0	WRM	0	0	0	0	0	0	0	AR	PC	0	AS	0	AVE	REAL

The meaning of each bit of the condition register is as follows:

Bit 0	REAL	Set to 1 while the realtime measurement is in progress.
Bit 1	AVE	Set to 1 while the averaging measurement is in progress.
Bit 3	AS	Set to 1 while auto searching.
Bit 5	PC	Set to 1 while the plug is being checked.
Bit 6	AR	Set to 1 while the auto range measurement is in progress.
Bit 14	WRM	Set to 1 during warm-up.

4.5 Output Queue and Error Queue

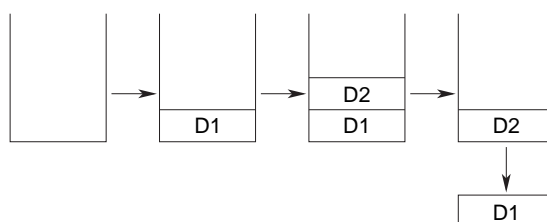
Overview of the Output Queue

The output queue is provided to store response messages to queries. For example, when the :WAVEform:SEND? query is sent to request output of the acquired waveform, the response data will be stored in the output queue until it is read out. The example below shows that data is stored record by record in the output queue, and is read out oldest item first, newest item last.

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 dead lock occurs (page 4-2)
- When a device clear command (DCL or SDC) is received
- When power is turned ON again

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.



Overview of the 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 number, "113, "Undefined header", and the error message are stored in the error queue, when the error is displayed. The contents of the error queue can be read using the :STATus:ERRor? query. As with the output queue, messages are read oldest first, newest last (refer to the previous page). If the error queue becomes full, the final message will be replaced by message "350, "Queue overflow".

The error queue is emptied in the following cases (in addition to when read-out is performed).

- When the *CLS command is received
- When power is turned ON again

To see whether the error queue is empty or not, check bit 2 (EAV) of the status byte.

4.6 Error Messages

Error in Execution

Code	Message
50	*OPC/? exists in message.
102	Syntax error.
103	Invalid separator.
104	Data type error.
108	Parameter not allowed.
109	Missing parameter.
111	Header separator error.
112	Program mnemonic too long.
113	Undefined header.
114	Header suffix out of range.
120	Numeric data error.
123	Exponent too large.
124	Too many digits.
128	Numeric data not allowed.
131	Invalid suffix.
134	Suffix too long.
138	Suffix not allowed.
141	Invalid character data.
144	Character data too long.
148	Character data not allowed.
150	String data error.
151	Invalid string data.
158	String data not allowed.
161	Invalid block data.
168	Block data not allowed.
171	Invalid expression.
178	Expression data not allowed.
181	Invalid outside macro definition.
210	Communication not allowed during trace analysis
211	Communication not allowed during measurement.
221	Setting conflict.
222	Data out of range.
223	Data invalid
224	Illegal parameter value.
241	Hardware missing.
260	Expression error.
270	Macro error.
272	Macro execution error.
273	Improper macro label.
275	Macro definition too long.
276	Macro recursion error.
277	Macro redefinition not allowed.
278	Macro header not found.
350	Queue overflow.
410	Query INTERRUPTED.
420	Query UNTERMINATED.
430	Query DEADLOCKED.
440	Query UNTERMINATED after indefinite response.

Code	Message
500	The measurement condition is failed. Please reset ***** in SETTING menu.
501	Not executable during measurement. Please stop the measurement and execute again.
502	The measurement cannot be started during printing. Please either wait until the end of the printing or interrupt the printing.
503	The measurement cannot be started during the file operation. Please either interrupt the file operation or wait until the end of the file operation.
504	The measurement could not be completed within the specified averaging duration. Please modify the averaging duration.
505	Hardware failed, and needs to be repaired. Please contact Yokogawa's representatives.
506	Hardware failed, and needs to be repaired. Please contact Yokogawa's representatives.
507	Hardware failed, and needs to be repaired. Please contact Yokogawa's representatives.
508	Hardware failed, and needs to be repaired. Please contact Yokogawa's representatives.
509	Hardware failed, and needs to be repaired. Please contact Yokogawa's representatives.
560	When the event fix is ON, the distance reference setup cannot be performed.
800	This function is not supported.
814	Failed network initialize. Please confirm network connection and setting.
816	Network setting is updated. Rebooting is required to reflect the changes.
817	Test Error occurred.
818	Test succeeded.
900	Backup data is damaged. this instrument starts up with the factory setting.
901	Optical connector is not correctly connected. Please clean and reconnect the connector again.
909	The temperature inside the instrument is too high. The instrument may be damaged if it is kept using in this condition. The instrument will be powered off in 10 sec. Please do not power it on till the battery is cooled down.
910	The temperature inside the instrument is too low. The instrument may be damaged if it is kept using in this condition. The instrument will be powered off in 10 sec. Please do not power it on till the battery is cooled down.
911	The Voltage of AC power supply is too low. The instrument may be damaged if it is kept using in this condition. The instrument will be powered off in 10 sec. Please make sure of using the dedicated AC adapter.
912	The Voltage of AC power supply is too high. The instrument may be damaged if it is kept using in this condition. The instrument will be powered off in 10 sec. Please make sure of using the dedicated AC adapter.
917	The temperature inside the instrument is increasing. Please stop measuring, and wait for a while.
918	The temperature inside the instrument is too high. The measurement was aborted. Please power it off.
921	There is incoming light. Or, optical module may be damaged.
922	Incorrect date and time setting. Set the correct date and time.
923	The ambient temperature is out of range. The light source cannot be turned on.
925	Please use AC adapter.