User's Manual

AQ1200A/AQ1200B/AQ1200C/ AQ1200E/AQ1205A/AQ1205E/ AQ1205F OTDR Multi Field Tester Communication Interface

YOKOGAWA
Yokogawa Test & Measurement Corporation

IM AQ1200-17EN
6th Edition

Thank you for purchasing the AQAQ1200A/AQ1200B/AQ1200C/AQ1200E/AQ1205A/AQ1205E/AQ1205F OTDR Multi Field Tester (hereinafter, “AQ1200” will refer to these products). This Communication Interface User’s Manual describes the functions and commands of the following communication interfaces.

• USB Interface

• Ethernet Interface (Optional)

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 AQ1200. Please read all of them.

List of Manuals

Manual Title	Manual No.	Description
AQ1200A/AQ1200B/AQ1200C/AQ1200E/ AQ1205A/AQ1205E/AQ1205F OTDR Multi Field Tester Operation Guide	IM AQ1200-02EN	This guide focuses on the handling precautions, basic operations, and specifications of the AQ1200.
AQ1200A/AQ1200B/AQ1200C/AQ1200E/ AQ1205A/AQ1205E/AQ1205F OTDR Multi Field Tester User’s Manual (included in CD)	IM AQ1200-01EN	Explains all AQ1200 features, except for the communication features, and how to use them.
AQ1200A/AQ1200B/AQ1200C/AQ1200E/ AQ1205A/AQ1205E/AQ1205F OTDR Multi Field Tester Communication Interface User’s Manual (included in CD)	IM AQ1200-17EN	This manual. Explains the features related to using communication commands to control the AQ1200.
Battery Pack (MFT) Handling Precautions	IM 739882-01EN	This document explains the handling precautions of the battery pack.
AQ1200A/AQ1200B/AQ1200C/AQ1200E/ AQ1205A/AQ1205E/AQ1205F OTDR User’s Manual	IM AQ1200-92Z1	A document for China.

The “-EN” in the manual number is the language code.

Contact information of Yokogawa offices worldwide is provided on the following sheet.

Document No.	Description
PIM 113-01Z2	List of worldwide contacts

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.
- The TCP/IP software of this product and the documents concerning it have been developed/created by YOKOGAWA based on the BSD Networking Software, Release 1 that has been licensed from the Regents of the University of California.

Trademarks

- Microsoft, Windows, Windows XP and Windows Vista, Windows 7, Windows 8 are either registered trademarks or trademarks of Microsoft Corporation in the United States and/or other countries.
- Adobe and Acrobat are trademarks of Adobe Systems Incorporated.
- In this manual, the TM and ® symbols do not accompany their respective registered trademark or trademark names.
- Other company and product names are registered trademarks or trademarks of their respective holders.

6th Edition : October 2017 (YMI)

Revisions

February 2010: 1st Edition
February 2011: 2nd Edition
July 2012: 3rd Edition
September 2015: 4th Edition
December 2015: 5th Edition
October 2017: 6th Edition

USB Interface and Ethernet Interface

- The items below are needed on the PC to use the communication functions via the USB interface.
- The communication libraly (TMCTL)
- USB driver
- The item below is needed on the PC to use the communication functions via the Ethernet interface.
- The communication libraly (TMCTL)

The library and driver above can be downloaded from the following Web page.
http://www.yokogawa.com/tm/

How to Use this Manual

Structure of the Manual

This User's Manual consists of the following sections:

Chapter 1 USB Interface

Describes the functions and specifications of the USB interface used to control the AQ1200 OTDR Multi Field Tester from a PC.

Chapter 2 Ethernet Interface (Option)

Describes the functions and specifications of the Ethernet interface.

Chapter 3 Before Programming

Describes the syntax used to transmit commands.

Chapter 4 Commands

Describes each command that is available.

Chapter 5 Condition Register / Output Queue and Error Queue

Describes the register and queues.

Appendix

Explains the support for AQ1200 error cord.

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.

Terms Used in Explanations of Procedures

Panel Keys and Soft Keys
Bold characters used in the procedural explanations indicate characters that are marked on the panel keys or the characters of the soft keys displayed on the screen menu.

Units

k: Denotes “1000.” Example: 400km
K: Denotes “1024.” Example: 459 KB (file data size)

Symbols Used in Syntax Descriptions

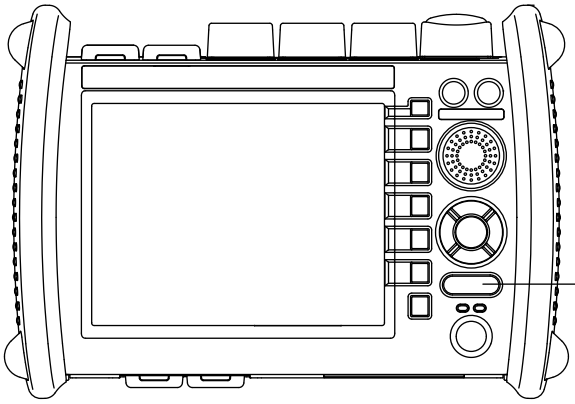
Symbols which are used in the syntax descriptions in Chapter 4 are shown below.
These symbols are referred to as BNF notation (Backus-Naur Form).
For detailed information, see section 3.4, “Data.”

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		

Contents

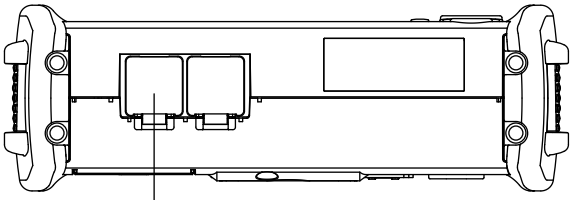
1.1 Names and Functions of Parts

Front Panel



SETUP key
Press this key to select the USB interface.

Bottom



USB type B (mini B) connector
Connector used to connect the AQ1100 to the controller (such as a PC) using a USB cable.

1.2 USB Interface Functions and Specifications

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 data, setup data of the panel, and error codes.

Transmission Function

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

USB Interface Specifications

Electrical and mechanical specifications:	Conforms to USB Rev.1.1
Connector:	Type B (mini B) connector (receptacle)
Number of ports:	1
Power supply:	Self-powered
Compatible PC systems:	PC running Windows 2000, Windows XP, or Windows Vista with a standard USB port (a separate device driver is needed to connect to a PC).

Switching between Remote and Local Modes

When Switching from Local to Remote Mode

Sending a command when the instrument is in the local mode causes the instrument to switch to the remote mode.

- All keys except the **Local** soft key are disabled.
- Settings entered in local mode are retained even when the AQ1200 switches to remote mode.

When Switching from Remote to Local Mode

Pressing the **Local** soft key in remote mode puts the instrument in local mode.

- Key operations are enabled.
- Settings entered in remote mode are retained even when the AQ1200 switches to local mode.

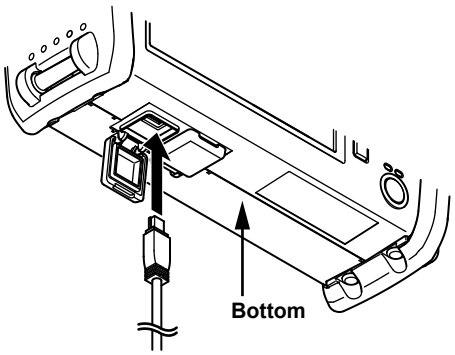
Note

The AQ1200 cannot be remotely controlled via the USB interface while the storage function is in operation. Remote control via the Ethernet interface is also not possible.

1.3 Connecting via the USB Interface

Connection Procedure

1. Open the bottom cover of the left side.
2. Connect a USB cable to the type B (mini 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 AQ1200 series is ready for operation (approximately 20 s).

1.4 Setting the AQ1200 (USB)

Procedure

Selecting the USB Interface Function

1. Press **SETUP** and then the **System Setup** soft key to display the following screen.

Set the mode to Control I/O.

Releasing the Remote Control

1. Press the **Local** soft key that appears on the screen after communication starts.

Explanation

USB Interface

To control the AQ1200 remotely using communication commands through the USB port, 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.

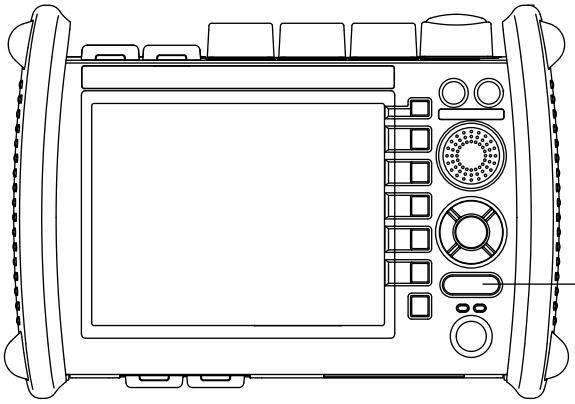
http://www.yokogawa.com/tm/tm-softdownload.htm

Note

- You cannot change the display while the AQ1200 is being remotely controlled.
- Only use the USB TMC driver (or software) provided by YOKOGAWA.

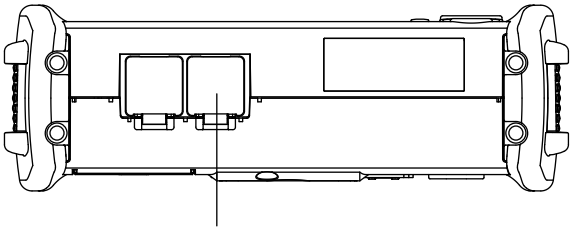
2.1 Names and Functions of Parts

Front Panel



SETUP key
Press this key to select the ethernet interface.

Bottom



Ethernet Port
Connector used to connect the AQ1100 to the controller (such as a PC) using a ethernet cable.

2.2 Ethernet Interface Functions and Specifications

Ethernet Interface Features

Reception Feature

The AQ1200 reception feature allows you to specify the same settings through an Ethernet connection that you can specify using the front panel keys.
The AQ1200 can receive output requests for measured data, panel setting data, and error codes.

Transmission Feature

The AQ1200 can transmit measured data.
The AQ1200 can transmit panel setting data and the status byte.
The AQ1200 can transmit error codes when errors occur.

Ethernet Interface Specifications

Electrical and mechanical specifications: Conforms to IEEE802.3
Transmission system: Ethernet (10BASE-T/100BASE-TX)
Data rate: 10 Mbps/100 Mbps
Number of communication ports: 1
Port number: 10001/tcp
Communication protocol: VXI-11
Connector type: RJ45 connector

Switching between Remote and Local Modes

When Switching from Local to Remote Mode

Sending a command when the instrument is in the local mode causes the instrument to switch to the remote mode.

- All keys except the **Local** soft key are disabled.
- Settings entered in local mode are retained even when the AQ1200 switches to remote mode.

When Switching from Remote to Local Mode

Pressing the **Local** soft key in remote mode puts the instrument in local mode.

- Key operations are enabled.
- Settings entered in remote mode are retained even when the AQ1200 to local mode.

Note

The AQ1200 cannot be remotely controlled via the USB interface while the storage function is in operation. Remote control via the Ethernet interface is also not possible.

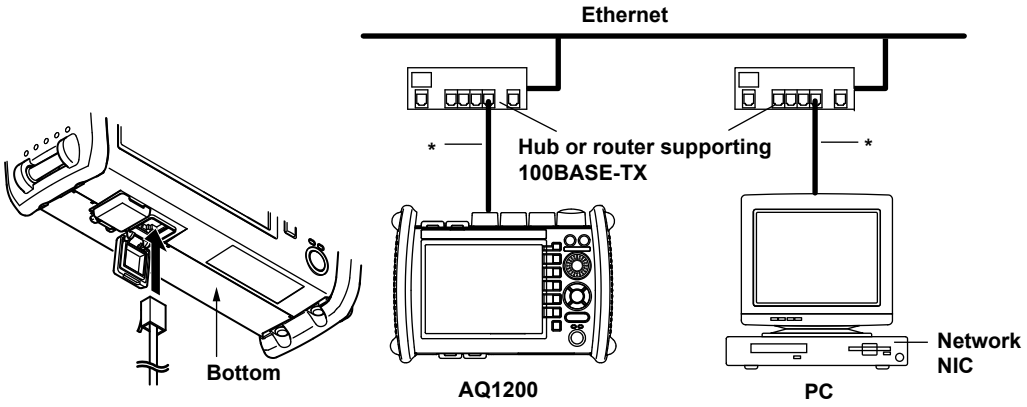
FTP Function

The AQ1200 series has an FTP function. You can transfer the data stored in the AQ1200 internal memory to the PC using FTP commands from the PC.

2.3 Connecting the Ethernet Interface

Connection Procedure

1. Open the bottom cover of the right side.
2. 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 on the bottom of the AQ1200.



* UTP cable or STP cable (straight cable in either case)

Precautions to Be Taken When Connecting the Cable

- Be sure to use a straight cable via a hub for the connection between the AQ1200 and the PC.
- Use a network cable that conforms to your network environment (such as the data rate).
- When using a UTP cable (straight cable), use a cable of category 5.

2.4 Setting the AQ1200 (Network)

The settings for remotely controlling the AQ1200 via the Ethernet interface are explained below.

Procedure

Selecting the Ethernet Interface Function

1. Press **SETUP** and then the **System Setup** soft key to display the following screen.

Set the Ethernet Interface.

Releasing the Remote Control

1. Press the **Local** soft key that appears on the screen after communication starts.

Explanation

Setting the Ethernet Interface

You must set the following parameters to use the Ethernet interface function.

- **Enabling or Disabling the Network Setup**
  After you have set the user name, password, timeout value, and TCP/IP parameters, select Valid and then restart the AQ1200 to use the network connection.
- **User Name**
- **Password**
- **Setting the Timeout Value**
  The connection to the network is automatically disconnected if there is no access to the AQ1200 for the specified time.
- **Setting the TCP/IP**
  IP Address
  Subnet Mask
  Default Gateway

For details on how to configure the settings, see section 19.7, “Configuring Network Settings (Option),” in the *AQ1200 OTDR Multi Field Tester User's Manual*, IM AQ1200-01EN.

Note

- You must restart the AQ1200 if you change the Ethernet settings. Before you restart the AQ1200, the settings from before you changed the settings are used.
- The AQ1200 cannot be remotely controlled via the ethernet interface while the USB storage function is in operation.

3.1 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

Program Message Unit

A program message consists of zero 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 (;).

Example `:PMETer:MODulation MOD_CW;UNIT DB<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" 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 added (NL is not included in the program message unit.)

Program Header

A program header is used to indicate the command type. For details, see section 3.2, “Commands.”

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 section 3.4, “Data.”

Example `:PMETer:MODulation MOD_CW<PMT>`

Header Data

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 `:PMETer:MODulation MOD_CW;UNIT DB<PMT>`

Unit Unit

<RMT>

RMT stands for “response message terminator.” The response message terminator is NL^EOM.

Response Header

A response header sometimes precedes the response data. A space separates the data from the header. For details, see section 3.3, “Response.”

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 section 3.4, “Data.”

Example `850E-9<RMT> :PMETer:MODulation MOD_CW<PMT>`

Data Header Data

If there are multiple queries in a program message, responses are returned in the same order that the queries were received in. The AQ1200 returns a single response message unit to most queries, but there are queries that the AQ1200 returns multiple units to. The first response message unit always corresponds to the first query, but the n^{th} response unit may not necessarily correspond to the n^{th} query. If you want to make sure that every response is retrieved, divide the program messages into individual messages.

Note

- 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 receivedwill be discarded.
- If an attempt is made by the controller to receive a response message, even if there it 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

Command

There are three types of commands (program headers) that a controller may send to the AQ1200. The commands differ in their program header formats.

Common Command Header

Commands that are defined in IEEE 488.2-1987 are called common commands. Be sure to include an asterisk (*) at the beginning of a common command.

Common command example `*CLS`

Compound Header

Other commands that are specific to the AQ1200 are classified and arranged in a hierarchy according to their functions. Be sure to use a colon to specify a lower hierarchical level.

Compound header example `:PMETer:LINK:STATe`

When Concatenating Commands

Command Groups

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

Commands relating to acquisition settings

```
:PMETer:DREF
:PMETer:LINK:STATe
:PMETer:MAXMin:STATe
:PMETer:MODulation
:PMETer:OFFSet
:PMETer:REFerence
:PMETer:WAVelength:DETail
```

When Concatenating Commands of the Same Group

The AQ1200 stores the hierarchical level of the command that is currently being executed and processes the next command on the assumption that it belongs to the same level. Therefore, the common header section can be omitted for commands that belong to the same group.

Example

```
:PMETer:MODulation MOD_CW;UNIT DB<PMT>
```

When Concatenating Commands of Different Groups

If the subsequent command does not belong to the same group, place a colon in front of the header (cannot be omitted).

Example

```
:PMETer:MODulation MOD_CW;:MENU:
FUNCtion TOP<PMT>
```

When Concatenating Common Commands

Common commands that are defined in the IEEE 488.2-1987 are independent of hierarchy. There is no need to use a colon.

Example

```
:PMETer:MODulation MOD_CW;*CLS;UNIT
DB<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 commandeven when commands of the same command groupare being concatenated.

Example

```
:PMETer:MODulation MOD_CW<PMT>:PMETer
UNIT DB<PMT>
```

Upper-level Query

An upper-level query is a query that is made by appending a question mark to the highest level command of a group. The controller can receive all of the settings in a group collectively by executing an upper-level query. Some upper-level queries of a group, which may be comprised of more than three hierarchical levels, can cause the AQ1200 to transmit all the lower level settings.

Example

```
:NETWork:CONTrol?<PMT> -> :NETW:CONT:
PASS "ABC";TIM 30;USER "anonyumous"
```

Note

- The response to an upper-level query can be sent back to the AQ1200 as a program message. This enables the settings that were present when the upper-level query was made to be reproduced later on.
- Some upper-level queries do not return setup data that is not currently in use. Exercise caution because not all of a group's information is necessarily returned in a response.

Header Interpretation Rules

The AQ1200 interprets the header that it receives according to the rules below.

Example

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

- The lower-case characters can be omitted.

  Example

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

- The question mark at the end of a header indicates that it is a query. You cannot omit the question mark.

  Example

  The shortest abbreviation for "DRANge?" is "DRAN?."

- If the <x> (value) at the end of a mnemonic is omitted, it is interpreted as a 1.

  Example

  If you write "M" for "M<x>, " "M1." is specified.

Note

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

3.3 Response

Form

When the controller sends a query with a question mark, the AQ1200 returns a response message to the query.

Response Consisting of a Header and Data

Responses that can be used as program messages without any changes are returned with command headers attached.

Example

```
:PMETer:MODulation?<PMT>
-> :PMETer:MODulation MOD_270HZ<RMT>
```

If You Want the AQ1200 to Return Responses without Headers

You can configure the AQ1200 so that even responses that have both headers and data are returned without headers. Use the COMMunicate:HEADer command for this purpose.

Abbreviated Form

The AQ1200 normally returns response headers with the lower-case section removed. You can configure the AQ1200 so that full headers are returned. Use the COMMunicate:VERBose command for this purpose.

3.4 Data

Data contains conditions and values that are written after the header. A space separates the data from the header. Data is grouped as follows:

Data	Description
<Decimal>	A value expressed in decimal notation (Example: Western calendar year `->MISC:DATE:YEAR 2009`)
<Distance><Time> <Wavelength><Loss>	A physical value (Example: Timeout value `->NETWork:CONTrol:TIMeout 30`))
<Character data>	Predefined character string (mnemonic). Select from the available strings in braces. (Example: Select the function mode `->MENU:FUNCtion {TOP|LSPM|LOSStest|PONPm|MLOSstest|IPTest}`)
<Boolean>	Indicates ON and OFF. Specify ON or OFF. (Example: Turn on the DHCP `->NETWork:DHCP ON`))
<String data>	User-defined string (Example: Set the Network password `->NETWork:CONTrol:PASSword "ABC"`)

<Decimal>

<Decimal> indicates a value expressed as a decimal number, as shown in the table below. Decimal values are written in the NR form as 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.			

- The AQ1200 can receive decimal values that are sent from the controller in any form, from <NR1> to <NR3>. This is expressed as <NRf>.
- The AQ1200 returns a response to the controller in one of the forms from <NR1> to <NR3> depending on the query. The same form is used regardless of the size of the value.
- For the <NR3> form, the plus sign after the “E” can be omitted. You cannot omit the minus sign.
- If a value outside the setting range is entered, the value is adjusted to the closest value within the range.
- If a value has more significant digits than are available, the value will be rounded.

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

<Distance>, <Time>, <Wavelength>, and <Loss> indicate decimal values that have physical significance. A <Multiplier> or <Unit> can be attached to the <NRf> form that was described earlier. The following types of expressions are possible.

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

<Multiplier>

<Multipliers> that you can use are indicated in the following table.

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> that you can use are indicated in the following table.

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 micro (“μ”).
- “MA” is used for Mega to distinguish it from Milli.
- If both <Multiplier> and <Unit> are omitted, the default unit is used.

<Character Data>

<Character data> is a predefined character string (mnemonics). It is mainly used to indicate that an option listed as a character string in braces must be selected and entered. The data interpretation rules are the same as those described in “Header Interpretation Rules” on page 3-4.

Form	Example
{SIMPLE|DETAIL|WIZARD|MULTI}	DETAIL

- 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 <character data>.

<Boolean>

<Boolean> is data that indicates ON or OFF. The following types of expressions are possible.

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 is selected for all other cases.
- A response message is always returned with a 1 if the value is ON and with a 0 if the value is OFF.

<String Data>

<String data> is not a predefined character string like <character data>. It can be any character string. The character string must be enclosed in single quotation marks (') or double quotation marks (").

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

- If a character string contains a double quotation mark ("), the double quotation mark is expressed as two consecutive quotation marks (""). This rule also applies to single quotation marks.
- A response message is always enclosed in double quotation marks (").
- <String data> is any character string. Therefore, the AQ1200 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.

<Block data>

<Block data> is any 8-bit data. It is only used in response messages on the AQ1200. The syntax is as follows:

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

- #N
  Indicates that the data is <block data>.“N” indicates the number of succeeding data bytes (digits) in ASCII code.
- <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 included in the data. Hence, care must be taken when programming the controller.

4.1 List of Commands

Command	Function	Page
ACQuire Group		
:ACQuire:ADSave	Sets or queries whether or not a file is saved automatically after averaged measurement is executed.	4-7
:ACQuire:AESearch	Sets or queries whether or not automatic event searching takes place after averaged measurement is executed.	4-7
:ACQuire:ATTenuation	Sets or queries the attenuation.	4-7
:ACQuire:AUTO:ATTenuation?	Queries the attenuation value when the attenuation is set to auto.	4-7
:ACQuire:AUTO:DRANge?	Queries the distance range value when the distance range is set to auto.	4-7
:ACQuire:AUTO:PWIDth?	Queries the pulse width when the pulse width is set to auto.	4-7
:ACQuire:AVERage:CONTinue	Sets or queries whether or not averaging is continued.	4-7
:ACQuire:AVERage:COUNt?	Queries the current average count.	4-7
:ACQuire:AVERage:INDex	Sets or queries the average index.	4-7
:ACQuire:AVERage:MODE	Sets or queries the averaging mode.	4-7
:ACQuire:AVERage:STARt	Executes averaged measurement.	4-8
:ACQuire:AVERage:STOP	Stops averaged measurement.	4-8
:ACQuire:AVERage:TIME	Sets or queries the averaged time.	4-8
:ACQuire:AVERage:TYPE	Sets or queries the average type.	4-8
:ACQuire:DRANge	Sets or queries the distance range.	4-8
:ACQuire:OFFSet	Sets or queries the horizontal axis measurement start position.	4-8
:ACQuire:OPTical:PADJust	Sets or queries the optical power output level.	4-8
:ACQuire:PLUGcheck	Sets or queries the optical plug connection check feature.	4-8
:ACQuire:PON:STATe	Sets or queries the PON (Passive Optical Network).	4-9
:ACQuire:PWIDth	Sets or queries the pulse width.	4-9
:ACQuire:REALtime:STARt	Executes real-time measurement.	4-9
:ACQuire:REALtime:STOP	Stops real-time measurement.	4-9
:ACQuire:SETTing	Sets or queries the measurement mode.	4-9
:ACQuire:SMPinterval:DATA	Sets or queries the sampling interval.	4-9
:ACQuire:SMPinterval:VALue?	Queries the sampling interval.	4-9
:ACQuire:WAVelength	Sets or queries the measurement wavelength.	4-9
ANALysis Group		
:ANALysis:ASEarch:EXECute	Executes automatic searching.	4-10
:ANALysis:ASEarch:NUMber?	Queries the number of automatically detected events.	4-10
:ANALysis:BCOefficient	Sets or queries the backscattering coefficient of the current wavelength.	4-10
:ANALysis:CURSor:DECibel?	Queries the cursor dB value.	4-10
:ANALysis:CURSor:DELete	Deletes the cursor.	4-10
:ANALysis:CURSor:DISTance	Sets or queries the cursor position.	4-10
:ANALysis:CURSor:LINK	Sets or queries the cursor link.	4-10
:ANALysis:DUNit	Sets or queries the distance unit.	4-10
:ANALysis:EMARker:LMTechnique	Sets or queries the approximation method (for events).	4-10
:ANALysis:EMARker:SET:M1	Sets or queries marker M1 of the current event.	4-10
:ANALysis:EMARker:SET:M2	Sets or queries marker M2 of the current event.	4-10
:ANALysis:EMARker:SET:M3	Sets or queries marker M3 of the current event.	4-11
:ANALysis:EMARker:SET:Y2	Sets or queries marker Y2 of the current event.	4-11
:ANALysis:EVENt:CURRent:CUMLoss?	Sets or queries the cumulative loss of the current event.	4-11
:ANALysis:EVENt:CURRent:DISTance?	Acquires the distance of the current event.	4-11
ANALysis:EVENt:CURRent:INDex	Moves the current event.	4-11
:ANALysis:EVENt:CURRent:LOSS?	Acquires the connection loss of the current event.	4-11
:ANALysis:EVENt:CURRent:RETurnloss?	Acquires the optical return loss of the current event.	4-11
:ANALysis:EVENt:CURRent:TYPE?	Acquires the event type of the current event.	4-11
:ANALysis:EVENt:CURRent:UNITloss?	Acquires the dB/km of the current event.	4-11
:ANALysis:EVENt:DELete	Deletes the current event.	4-12
:ANALysis:EVENt:FIX:MODE	Sets or queries the event fix mode.	4-12

Command	Function	Page
:ANALysis:EVENt:FIX:STATe	Sets or queries the event fix on/off state.	4-12
:ANALysis:EVENt:INSert	Inserts an event at the cursor position.	4-12
:ANALysis:EVENt:IOR	Sets or queries the section group index of the current event.	4-12
:ANALysis:FEDetection	Sets or queries whether or not fault events are detected.	4-12
:ANALysis:FMARker:DELete	Deletes the marker.	4-12
:ANALysis:FMARker:LEFT:DISTance?	Queries the distance between markers 1 and 2.	4-12
:ANALysis:FMARker:LEFT:LOSS?	Queries the loss between markers 1 and 2.	4-12
:ANALysis:FMARker:LEFT:UNITloss?	Queries the slope between markers 1 and 2.	4-12
:ANALysis:FMARker:LMTechnique	Sets or queries the approximation method (for markers).	4-12
:ANALysis:FMARker:LOSS?	Queries the connection loss.	4-12
:ANALysis:FMARker:REFLection:SATurated?	Queries the reflection saturation.	4-12
:ANALysis:FMARker:REFLection:VALue?	Queries the amount of reflection.	4-12
:ANALysis:FMARker:RETurnloss:SATurated?	Queries the optical return loss saturation.	4-13
:ANALysis:FMARker:RETurnloss:VALue?	Queries the optical return loss.	4-13
:ANALysis:FMARker:RIGHt:DISTance?	Queries the distance between markers 2 and 3.	4-13
:ANALysis:FMARker:RIGHt:LOSS?	Queries the loss between markers 2 and 3.	4-13
:ANALysis:FMARker:RIGHt:UNITloss?	Queries the slope between markers 2 and 3.	4-13
:ANALysis:FMARker:SET:M<x>	Sets or queries markers.	4-13
:ANALysis:FMARker:SET:Y<x>	Sets or queries auxiliary markers.	4-13
:ANALysis:IOR	Sets or queries the index of refraction of the current wavelength.	4-13
:ANALysis:MACRobending:DISPlay	Sets or queries the macrobending display on/off state.	4-13
:ANALysis:MACRobending:THReshold	Sets or queries the macrobending threshold.	4-13
:ANALysis:REFerence:DELete	Deletes the distance reference.	4-13
:ANALysis:REFerence:DISTance	Sets the distance reference or queries its current position.	4-13
:ANALysis:THReshold:EOFiber	Sets or queries the fiber end threshold.	4-14
:ANALysis:THReshold:FERLoss	Sets or queries the fault event optical return loss threshold.	4-14
:ANALysis:THReshold:FESLoss	Sets or queries the fault event connection loss threshold.	4-14
:ANALysis:THReshold:RLOSs	Sets or queries the optical return loss threshold.	4-14
:ANALysis:THReshold:SLOSs	Sets or queries the connection loss threshold.	4-14
:ANALysis:SECTion:BASelevel?	Queries the dB value of the interval data reference point.	4-14
:ANALysis:SECTion:DELete	Deletes the interval analysis data.	4-14
:ANALysis:SECTion:DISTance?	Queries the distance of the interval data.	4-14
:ANALysis:SECTion:END	Sets or queries the end position of the interval data.	4-14
:ANALysis:SECTion:LMTechnique	Sets or queries the interval analysis approximation method.	4-14
:ANALysis:SECTion:LOSS?	Queries the loss of the interval data.	4-15
:ANALysis:SECTion:REFerence	Sets the interval data reference point.	4-15
:ANALysis:SECTion:RETurnloss:SATurated?	Queries the optical return loss saturation of the interval data.	4-15
:ANALysis:SECTion:RETurnloss:VALue?	Queries the optical return loss of the interval data.	4-15
:ANALysis:SECTion:STARt	Sets or queries the start position of the interval data.	4-15
:ANALysis:TRACefix:STATe	Sets or queries the snapshot (waveform freeze) feature.	4-15

COMMunicate Group

:COMMunicate?	Queries all communication settings.	4-16
:COMMunicate:HEADer	Sets or queries whether or not a header is added to the response to a query. (Example with header: PMETer:REFerence 5.00. Example without header: 5.00.)	4-16
:COMMunicate:VERBose	Sets or queries whether the response to a query is returned fully spelled out (example: PMETer:REFerence 5.00) or using abbreviation (example: PMET:REF 5.00).	4-16

Command	Function	Page
DISPlay Group		
:DISPlay:ALINe	Sets or queries the display of approximated lines.	4-17
:DISPlay:COLor	Sets or queries the screen colors.	4-17
:DISPlay:CURSor:DBValue	Sets or queries the cursor dB value.	4-17
:DISPlay:CURSor:SECond	Sets or queries the second-cursor display.	4-17
:DISPlay:DECibel:UPPer	Sets or queries the display start level.	4-17
:DISPlay:DISTance:LEFT	Sets or queries the display start distance.	4-17
:DISPlay:DIVide:DECibel	Sets or queries the vertical axis zoom.	4-17
:DISPlay:DIVide:DISTance	Sets or queries the horizontal axis zoom.	4-17
:DISPlay:EVENt:AZOom	Sets or queries the event auto zoom.	4-17
:DISPlay:INITialize:SCALe	Initializes the display scale.	4-18
:DISPlay:MARKer:INFormation	Sets or queries the display of marker information.	4-18
:DISPlay:SCALe	Sets or queries the scale display.	4-18
:DISPlay:TOOLtip	Sets or queries the tool tip display.	4-18
FILE Group		
:FILE:DELete:EXECute	Deletes files.	4-19
:FILE:DRIVe:FREE?	Queries the amount of free space on the current drive.	4-19
:FILE:DRIVe:SET	Sets or queries the current drive setting.	4-19
:FILE:FILE:EXISt?	Checks whether or not the specified file exists.	4-19
:FILE:FILE:GET?	Acquires the specified file.	4-19
:FILE:FILE:NAME	Specifies the file name used for file acquisition, file size acquisition, and file transfer.	4-19
:FILE:FILE:SIZE?	Acquires the size of the specified file.	4-19
:FILE:FOLDer:LIST?	Acquires a list of the contents of the current folder.	4-19
:FILE:FOLDer:MAKE	Creates a folder.	4-19
:FILE:FOLDer:PATH	Sets or queries the current folder name.	4-20
:FILE:LOAD:EXECute	Loads a file.	4-20
:FILE:SAVE:COMMent	Sets or queries the save comment.	4-20
:FILE:SAVE:EXECute	Saves the file.	4-20
:FILE:SAVE:ID	Sets or queries the save ID.	4-20
:FILE:SAVE:ITEM<x>	Sets or queries the saved items.	4-20
:FILE:SAVE:SEParator	Sets or queries the separator used when files are saved.	4-20
:FILE:SAVE:SUB	Sets or queries the sub number used when files are saved.	4-20
:FILE:SAVE:TYPE	Sets or queries the filename type used when files are saved.	4-21
:FILE:SOR:GET?	Acquires an SOR file image.	4-21
:FILE:SOR:VALid?	Queries whether the SOR file image is enabled.	4-21
:FILE:SUBFolder:LIST?	Acquires a list of the subfolders in the current folder.	4-21
:FILE:TYPE	Sets or queries the type of file that is saved.	4-21
LABel Group		
:LABel:CABLe:CODE	Sets or queries the cable code.	4-22
:LABel:CABLe:ID	Sets or queries the cable ID.	4-22
:LABel:COMPany	Sets or queries the company name.	4-22
:LABel:DFLag:CURRent	Sets or queries the current data flag.	4-22
:LABel:FIBer:ID	Sets or queries the fiber ID.	4-22
:LABel:FIBer:TYPE	Sets or queries the fiber type.	4-22
:LABel:LOCation:ORIGinating	Sets or queries the start position label.	4-22
:LABel:LOCation:TERMinating	Sets or queries the end position label.	4-22
:LABel:OPERator	Sets or queries the operator name.	4-22
LIGHtsource Group		
:LIGHtsource:ABORt	Turns the measurement light off.	4-23
:LIGHtsource:EXECute	Turns the measurement light on.	4-23
:LIGHtsource:MODulation	Sets or queries the light source modulation frequency.	4-23
:LIGHtsource:STATe	Turns the measurement light off or on.	4-23
:LIGHtsource:WAVelength	Sets or queries the light source wavelength.	4-23

Command	Function	Page
MENU Group		
:MENU:ERRor:CLEar	Clears the error dialog box.	4-24
:MENU:FUNCtion	Sets or queries the function mode.	4-24
:MENU:MARKer	Sets or queries the marker mode.	4-24
MISC Group		
:MISC:ALARmsound	Sets or queries the alarm sound.	4-25
:MISC:BACKlightoff	Sets or queries the backlight off setting when the AQ1200 is running on battery power.	4-25
:MISC:BRIGhtness:AC	Sets or queries the LCD brightness when the AC adapter is connected.	4-25
:MISC:BRIGhtness:BATTery	Sets or queries the LCD brightness when the AQ1200 is running on battery power.	4-25
:MISC:BACKlightoff	Sets or queries the backlight off setting when the AQ1200 is running on battery power.	4-25
:MISC:DATE:DAY	Sets or queries the day.	4-25
:MISC:DATE:GET?	Queries the date and time.	4-25
:MISC:DATE:HOUR	Sets or queries the hour.	4-25
:MISC:DATE:MINute	Sets or queries the minute.	4-25
:MISC:DATE:MODE	Sets or queries the date display type.	4-25
:MISC:DATE:MONTh	Sets or queries the month.	4-26
:MISC:DATE:SECond	Sets or queries the second.	4-26
:MISC:DATE:SET	Applies the date and time change.	4-26
:MISC:DATE:YEAR	Sets or queries the year.	4-26
:MISC:LANGuage	Sets or queries the language.	4-26
:MISC:POWersave:AC	Sets or queries the power-save setting when the AC adapter is connected.	4-26
:MISC:POWersave:BATTery	Sets or queries the power-save setting when the AQ1200 is running on battery power.	4-26
:MISC:RLOSsmode	Sets or queries the reflection display.	4-26
NETWork Group		
:NETWork:CONTrol:PASSword	Sets or queries the password.	4-27
:NETWork:CONTrol:TIMeout	Sets or queries the timeout value.	4-27
:NETWork:CONTrol:USERname	Sets or queries the user name.	4-27
:NETWork:DHCP	Sets or queries the DHCP on/off state.	4-27
:NETWork:GATeway	Sets or queries the gateway.	4-27
:NETWork:IPADdress	Sets or queries the IP address.	4-27
:NETWork:NETMask	Sets or queries the subnet mask.	4-28
:NETWork:STATe	Sets or queries the Ethernet on/off state.	4-28
PMETer Group		
:PMETer:AVERage:TIMes	Sets or queries the power meter average count.	4-29
:PMETer:DREF	Executes Dref on the power meter.	4-29
:PMETer:LINK:STATe	Sets or queries the light source power meter setting interlock.	4-29
:PMETer:MAXMin:STATe	Sets or queries the MAX and MIN display on/off state.	4-29
:PMETer:MAXMin:MAX?	Sets or queries the maximum value.	4-29
:PMETer:MAXMin:MIN?	Sets or queries the minimum value.	4-29
:PMETer:MEASurement:DATA?	Queries the power meter's measured results.	4-29
:PMETer:MODulation	Sets or queries the power meter modulation.	4-29
:PMETer:OFFSet	Sets or queries the power meter offset.	4-29
:PMETer:REFerence	Sets or queries the power meter reference value.	4-29
:PMETer:THReshold:LOWer	Sets or queries the power meter lower threshold value.	4-29
:PMETer:THReshold:UPPer	Sets or queries the power meter upper threshold value.	4-30
:PMETer:UNIT	Sets or queries the power meter display unit.	4-30
:PMETer:WAVelength:DETail	Sets or queries the wavelength when the wavelength mode is set to Detail.	4-30
:PMETer:ZERoset	Executes zero set on the power meter.	4-30

Command	Function	Page
PON Group		
:PON:AVERage:TIMes	Sets or queries the PON power meter average count.	4-31
:PON:DIRection	Sets or queries the PON power meter wavelength.	4-31
:PON:M1310:MEASurement:DATA?	Queries the measured results of the 1310 nm PON power meter.	4-31
:PON:M1310:OFFSet	Sets or queries the 1310 nm PON power meter offset.	4-31
:PON:M1310:THReshold:JUDGe?	Queries the threshold judgment results of the 1310 nm PON power meter.	4-31
:PON:M1310:THReshold:LOWer	Sets or queries the lower threshold value of the 1310 nm PON power meter.	4-31
:PON:M1310:THReshold:UPPer	Sets or queries the upper threshold value of the 1310 nm PON power meter.	4-31
:PON:M1490:MEASurement:DATA?	Queries the measured results of the 1490 nm PON power meter.	4-31
:PON:M1490:OFFSet	Sets or queries the 1490 nm PON power meter offset.	4-31
:PON:M1490:THReshold:JUDGe?	Queries the threshold judgment results of the 1490 nm PON power meter.	4-31
:PON:M1490:THReshold:LOWer	Sets or queries the lower threshold value of the 1490 nm PON power meter.	4-31
:PON:M1490:THReshold:UPPer	Sets or queries the upper threshold value of the 1490 nm PON power meter.	4-31
:PON:M1550:MEASurement:DATA?	Queries the measured results of the 1550 nm PON power meter.	4-32
:PON:M1550:OFFSet	Sets or queries the 1550 nm PON power meter offset.	4-32
:PON:M1550:THReshold:JUDGe?	Queries the threshold judgment results of the 1550 nm PON power meter.	4-32
:PON:M1550:THReshold:LOWer	Sets or queries the lower threshold value of the 1550 nm PON power meter.	4-32
:PON:M1550:THReshold:UPPer	Sets or queries the upper threshold value of the 1550 nm PON power meter.	4-32
:PON:UNIT	Sets or queries the PON power meter display unit.	4-32
:PON:ZERoset	Executes zero set on the PON power meter.	4-32
PRINt Group		
:PRINt:COLor	Sets or queries the print colors.	4-33
:PRINt:EXECute	Executes printing.	4-33
:PRINt:MAKer	Sets or queries the printer manufacturer.	4-33
SETup Group		
:SETup:ALLinit	Returns all the settings to their factory default values and formats the internal memory.	4-34
:SETup:INITialize	Returns all settings to their factory defaults.	4-34
STATus Group		
:STATus?	Queries all the settings for the communication status feature.	4-35
:STATus:CONDition?	Queries the contents of the condition register.	4-35
:STATus:ERRor?	Queries the error code and message information (top of the error queue).	4-35
:STATus:QENable	Sets or queries whether or not messages other than errors will be stored to the error queue (on/off).	4-35
:STATus:QMESsage	Sets or queries whether or not message information will be attached to the response to the STATus:ERRor? query (on/off).	4-35
SYSTem Group		
:SYSTem:REBoot	Restarts the AQ1200.	4-36
:SYSTem:SHUTdown	Shuts down the AQ1200.	4-36
VLS Group		
:VLS:ABORt	Turns the visible light source off.	4-37
:VLS:EXECute	Turns the visible light source on.	4-37
:VLS:STATe	Turns the visible light source on or off.	4-37
WAVedata Group		
:WAVedata:LENGth?	Queries the number of waveform data points.	4-38
:WAVedata:DISPlay:SEND:ASCii?	Queries the data of the displayed waveform in ASCII format.	4-38
:WAVedata:DISPlay:SEND:BINary?	Queries the data of the displayed waveform in binary format.	4-38
:WAVedata:OLDType:DISPlay:SEND?	Queries the data of the displayed waveform in Dot4 format.	4-38
:WAVedata:OLDType:SEND?	Queries the waveform data in Dot4 format.	4-38
:WAVedata:SEND:ASCii?	Queries the waveform data in ASCII format.	4-38
:WAVedata:SEND:BINary?	Queries the waveform data in block data (in binary format).	4-38
:WAVedata:SEND:SIZE	Sets or queries the number of acquired data points.	4-38
:WAVedata:SEND:STARt	Sets or queries the starting distance of the waveform data.	4-39

Command	Function	Page
Common Commands		
*CLS (Clear Status)	Clears all event status registers that are displayed in the status byte register summary.	4-40
*ESE (Standard Event Status Enable)	Sets or queries the standard event enable register.	4-40
*ESR? (Standard Event Status Register)	Queries and clears the standard event status register.	4-40
*IDN? (Identification)	Queries the instrument type and firmware version.	4-40
*OPT? (Option)	Queries the option information.	4-40
*RST (Reset)	Returns all the settings except the communication settings to their factory default values.	4-40
*SRE (Service Request Enable)	Sets or queries the service request enable register.	4-41
*STB? (Read Status Byte)	Queries the current status byte register value.	4-41
*TST? (Self Test)	Executes a self-test.	4-41

4.2 ACQuire Group

The commands in this group deal with waveform acquisition. You can make the same settings and queries that you can by using the front panel. However, you need to stop measurement before you change measurement conditions remotely. You cannot change the measurement conditions during real-time measurement or during averaged measurement. During measurement, only the `:REALtime:STOP` or `AVERage:STOP` command is valid.

:ACQuire:ADSave

Function Sets or queries whether or not a file is saved automatically after averaged measurement is executed.

```
:ACQuire:ADSave {<Boolean>}
:ACQuire:ADSave?
```

Example
```
:ACQUIRE:ADSAVE ON
:ACQUIRE:ADSAVE? -> :ACQUIRE:ADSAVE 1
```

:ACQuire:AESearch

Function Sets or queries whether or not automatic event searching takes place after averaged measurement is executed.

Syntax
```
:ACQuire:AESearch {<Boolean>}
:ACQuire:AESearch?
```

Example
```
:ACQUIRE:AESEARCH OFF
:ACQUIRE:AESEARCH?
-> :ACQUIRE:AESEARCH 0
```

:ACQuire:ATTenuation

Function Sets or queries the attenuation.

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

Example
```
:ACQUIRE:ATTENUATION 2.5
:ACQUIRE:ATTENUATION AUTO
:ACQUIRE:ATTENUATION?
-> :ACQUIRE:ATTENUATION 25.00
```

Description
- The range of values that you can set varies depending on the wavelength, distance range, and pulse width.
- You cannot use this command if the averaging mode is set to high reflection.
- You cannot use this command if the distance range is set to auto.

:ACQuire:AUTO:ATTenuation?

Function Queries the attenuation value when the attenuation is set to auto.

Syntax
```
:ACQuire:AUTO:ATTenuation?
```

Example
```
:ACQUIRE:AUTO:ATTENUATION?
-> :ACQUIRE:AUTO:ATTENUATION 0.00
```

:ACQuire:AUTO:DRANge?

Function Queries the distance range value when the distance range is set to auto.

Syntax
```
:ACQuire:AUTO:DRANge?
```

Example
```
:ACQUIRE:AUTO:DRANGE?
-> :ACQUIRE:AUTO:DRANGE 20000
```

Description The unit is meters.

:ACQuire:AUTO:PWIDth?

Function Queries the pulse width when the pulse width is set to auto.

Syntax
```
:ACQuire:AUTO:PWIDth?
```

Example
```
:ACQUIRE:AUTO:PWIDTH?
-> :ACQUIRE:AUTO:PWIDTH 100E-09
```

Description The unit is seconds.

:ACQuire:AVERage:CONTinue

Function Sets or queries whether or not averaging is continued.

Syntax
```
:ACQuire:AVERage:CONTinue{<Boolean>}
:ACQuire:AVERage:CONTinue?
```

Example
```
:ACQUIRE:AVERAGE:CONTINUE ON
:ACQUIRE:AVERAGE:CONTINUE?
-> :ACQUIRE:AVERAGE:CONTINUE 1
```

:ACQuire:AVERage:COUNt?

Function Queries the current average count.

Syntax
```
:ACQuire:AVERage:COUNt?
```

Example
```
:ACQUIRE:AVERAGE:COUNT?
-> :ACQUIRE:AVERAGE:COUNT 0
```

:ACQuire:AVERage:INDex

Function Sets or queries the average index.

Syntax
```
:ACQuire:AVERage
:INDex{AUTO|E2_10|E2_11…….E2_20}
:ACQuire:AVERage:INDex?
```
AUTO: Automatic
E2_10:2E+10
E2_11:2E+11
• •
• •
E2_20:2E+20

Example
```
:ACQUIRE:AVERAGE:INDEX AUTO
:ACQUIRE:AVERAGE:INDEX E2_12
:ACQUIRE:AVERAGE:INDEX?
-> :ACQUIRE:AVERAGE:INDEX E2_15
```

:ACQuire:AVERage:MODE

Function Sets or queries the averaging mode.

Syntax
```
:ACQuire:AVERage:MODE
{HIREFLECTION|HISPEED}
:ACQuire:AVERage:MODE?
```
HIREFLECTION: High reflection
HISPEED: High speed

Example
```
:ACQUIRE:AVERAGE:MODE HIREFLECTION
:ACQUIRE:AVERAGE:MODE?
-> :ACQUIRE:AVERAGE:MODE HISPEED
```

:ACQuire:AVERage:STARt

Function Executes averaged measurement.

Syntax :ACQuire:AVERage:STARt

Example :ACQUIRE:AVERAGE:START

Description You can only use this command when averaged measurement is stopped.

:ACQuire:AVERage:STOP

Function Stops averaged measurement.

Syntax :ACQuire:AVERage:STOP

Example :ACQuire:AVERage:STOP

Description You can only use this command when averaged measurement is taking place.

:ACQuire:AVERage:TIME

Function Sets or queries the averaged time.

Syntax :ACQuire:AVERage:TIME {<NRf>|AUTO}
:ACQuire:AVERage:TIME?
<NRf> = 5, 10, 20, 30, 60, 180, 300, 600, 1200, 1800 (s)
AUTO: Automatic
5: 5 s
10: 10 s
20: 20 s
30: 30 s
60: 1 min
180: 3 min
300: 5 min
600: 10 min
1200: 20 min
1800: 30 min

Example :ACQUIRE:AVERAGE:TIME AUTO
:ACQUIRE:AVERAGE:TIME 1200
:ACQUIRE:AVERAGE:TIME?
-> :ACQUIRE:AVERAGE:TIME 60

:ACQuire:AVERage:TYPE

Function Sets or queries the average type.

Syntax :ACQuire:AVERage
:TYPE {TIMES|DURATION}
ACQuire:AVERage:TYPE?
TIMES: Times
DURATION: Duration

Example :ACQUIRE:AVERAGE:TYPE TIMES
:ACQUIRE:AVERAGE:TYPE DURATION
:ACQUIRE:AVERAGE:TYPE?
-> :ACQ:AVER:TYPE DURATION

:ACQuire:DRANge

Function Sets or queries the distance range.

Syntax :ACQuire:DRANge {<NRf>|AUTO}
:ACQuire:DRANge?
<NRf> = 500 m to 512000 m (0.5 km to 512 km)
AUTO: Automatic
500: 500 m
1000: 1 km
2000: 2 km
5000: 5 km
10000: 10 km
20000: 20 km
50000: 50 km
100000: 100 km
200000: 200km
300000: 300km
400000: 400km
512000: 512km

Example :ACQUIRE:DRANGE AUTO
:ACQUIRE:DRANGE 500
:ACQUIRE:DRANGE 500m
:ACQUIRE:DRANGE 5000
:ACQUIRE:DRANGE 100km
:ACQUIRE:DRANGE?
-> :ACQUIRE:DRANGE 1000

Description The values that you can set vary depending on the wavelength.

:ACQuire:OFFSet

Function Sets or queries the horizontal axis measurement start position.

Syntax :ACQuire:OFFSet {<NRf>}
:ACQuire:OFFSet?

Example :ACQUIRE:OFFSET 1000
:ACQUIRE:OFFSET?
-> :ACQUIRE:OFFSET 999.99265E+00

Description The unit is meters.

:ACQuire:OPTical:PADJust

Function Sets or queries the optical power output level.

Syntax :ACQuire:OPTical:
PADJust {LOW|NORMal>}
:ACQuire:OPTical:PADJust?

Example :ACQUIRE:OPTICAL:PADJUST NORMAL
:ACQUIRE:OPTICAL:PADJUST? -> :
ACQUIRE:OPTICAL:PADJUST NORMAL

Description You cannot use this command on the AQ1200A and AQ1205A.

:ACQuire:PLUGcheck

Function Sets or queries the optical plug connection check feature.

Syntax :ACQuire:PLUGcheck {<Boolean>}
:ACQuire:PLUGcheck?

Example :ACQUIRE:PLUGCHECK ON
:ACQUIRE:PLUGCHECK?
-> :ACQUIRE:PLUGCHECK 0

:ACQuire:PON:STATe

Function Sets or queries the PON (Passive Optical Network).

Syntax
```
:ACQuire:PON:STATe {<Boolean>}
:ACQuire:PON:STATe?
```

Example
```
:ACQUIRE:PON:STATE ON
:ACQUIRE:PON:STATE? -> :ACQUIRE:PON:
STATE 0
```

Description
- You can only use this command on the AQ1200A with the /PN option, AQ1200B, and AQ1200C.
- The PON can only be set when the measurement mode is Detail.

:ACQuire:PWIDth

Function Sets or queries the pulse width.

Syntax
```
:ACQuire:PWIDth {<NRf>|AUTO}
:ACQuire:PWIDth?
<NRf> = 3 ns to 20 us (3E-9 to 20E-6)
AUTO:    Automatic
3E-09:   3 ns
10E-09:  10 ns
20E-09:  20 ns
50E-09:  50 ns
100E-09: 100 ns
200E-09: 200 ns
500E-09: 500 ns
1E-06:   1 µs
2E-06:   2 µs
5E-06:   5 µs
10E-06:  10 µs
20E-06:  20 µs
```

Example
```
:ACQUIRE:PWIDTH AUTO
:ACQUIRE:PWIDTH 3E-9
:ACQUIRE:PWIDTH 3ns
:ACQUIRE:PWIDTH 10us
:ACQUIRE:PWIDTH 10E-6
:ACQUIRE:PWIDTH?
-> :ACQ:PWID 10.0E-06
```

Description
- The values that you can set vary depending on the wavelength and distance range.
- You cannot use this command if the distance range is set to auto.

:ACQuire:REALtime:STARt

Function Executes real-time measurement.

Syntax `:ACQuire:REALtime:STARt`

Example `:ACQUIRE:REALTIME:START`

Description You can only use this command when measurement is stopped.

:ACQuire:REALtime:STOP

Function Stops real-time measurement.

Syntax `:ACQuire:REALtime:STOP`

Example `:ACQUIRE:REALTIME:STOP`

Description You can only use this command when real-time measurement is taking place.

:ACQuire:SETTing

Function Sets or queries the measurement mode.

Syntax
```
:ACQuire:SETTing {SIMPLE|DETAIL
|MULTI}
:ACQuire:SETTing?
SIMPLE: Simple mode
DETAIL: Detailed mode
MULTI : Multi-wavelength measurement mode
```

Example
```
:ACQUIRE:SETTING DETAIL
:ACQUIRE:SETTING?
-> :ACQUIRE:SETTING SIMPLE
```

:ACQuire:SMPinterval:DATA

Function Sets or queries the sampling interval.

Syntax
```
:ACQuire:SMPinterval
:DATA {<NRf>|NORMAL|HI}
:ACQuire:SMPinterval:DATA?
NORMAL: Normal
HI:     High resolution
0.05: 5 cm
0.10: 10 cm
0.20: 20 cm
0.50: 50 cm
1.00: 1 m
2.00: 2 m
4.00: 4 m
8.00: 8 m
16.00: 16 m
32.00: 32 m
```

Example
```
:ACQUIRE:SMPINTERVAL:DATA 2.0
:ACQUIRE:SMPINTERVAL:DATA NORMAL
:ACQUIRE:SMPINTERVAL:DATA?
-> :ACQUIRE:SMPINTERVAL:DATA HI
```

Description When the distance range is set to auto, you can only set the interval to NORMAL or HI.

:ACQuire:SMPinterval:VALue?

Function Queries the sampling interval.

Syntax `:ACQuire:SMPinterval:VALue?`

Example
```
:ACQUIRE:SMPINTERVAL:VALUE?
-> :ACQUIRE:SMPINTERVAL:VALUE 8.0
```

:ACQuire:WAVelength

Function Sets or queries the measurement wavelength.

Syntax
```
:ACQuire:WAVelength {<NRf>}
:ACQuire:WAVelength?
<NRf> = 0.850 um to 1.650 um
(850E-9 to 1650E-9)
1310E-09: 1310 nm
1550E-09: 1550 mn
```

Example
```
:ACQUIRE:WAVELENGTH 0.85um
:ACQUIRE:WAVELENGTH 1650E-9
:ACQUIRE:WAVELENGTH 1.650E-6
:ACQUIRE:WAVELENGTH?
-> :ACQUIRE:WAVELENGTH 1550E-09
```

Description The wavelengths that can be specified vary depending on the model.

4.3 ANALysis Group

The commands in this group deal with waveform analysis. You can make the same settings and queries that you can by using the front panel.

:ANALysis:ASEarch:EXECute

Function Executes automatic searching.
Syntax :ANALysis:ASEarch:EXECute
Example :ANALYSIS:ASEARCH:EXECUTE
Description After execution, wait until the AS flag in the return value of the STATus:CONDition? command becomes zero, and then query the event results.

:ANALysis:ASEarch:NUMber?

Function Queries the number of automatically detected events.
Syntax :ANALysis:ASEarch:NUMber?
Example :ANALYSIS:ASEARCH:NUMBER?
-> :ANALYSIS:ASEARCH:NUMBER 2

:ANALysis:BCOefficient

Function Sets or queries the backscattering coefficient of the current wavelength.
Syntax :ANALysis:BCOefficient {<NRf>}
:ANALysis:BCOefficient?
<NRf> = -64.99 to -10.00 (in steps of 0.01)
Example :ANALYSIS:BCOEFFICIENT -25.00
:ANALYSIS:BCOEFFICIENT?
-> :ANALYSIS:BCOEFFICIENT -50.00

:ANALysis:CURSor:DECibel?

Function Queries the cursor dB value.
Syntax :ANALysis:CURSor:DECibel?
Example :ANALYSIS:CURSOR:DECIBEL?
-> :ANALYSIS:CURSOR:DECIBEL 32.878

:ANALysis:CURSor:DELete

Function Deletes the cursor.
Syntax :ANALysis:CURSor:DELete
Example :ANALYSIS:CURSOR:DELETE

:ANALysis:CURSor:DISTance

Function Sets or queries the cursor position.
Syntax :ANALysis:CURSor:DISTance {<NRf>}
:ANALysis:CURSor:DISTance?
<NRf> = 0 to distance range (m)
Example :ANALYSIS:CURSOR:DISTANCE 10.00
:ANALYSIS:CURSOR:DISTANCE?
-> :ANALYSIS:CURSOR:DISTANCE
11.529900E+00
Description
- Depends on the distance range, distance reference, and index of refraction.
- The unit is meters.

:ANALysis:CURSor:LINK

Function Sets or queries the cursor link.
Syntax :ANALysis:CURSor:LINK {<Boolean>}
:ANALysis:CURSor:LINK?
Example :ANALYSIS:CURSOR:LINK ON
:ANALYSIS:CURSOR:LINK?
-> :ANALYSIS:CURSOR:LINK 1

:ANALysis:DUNit

Function Sets or queries the distance unit.
Syntax :ANALysis:DUNit {KM|MILE|KF}
:ANALysis:DUNit?
KM: km
MILE: mile
KF: kf
Example :ANALYSIS:DUNIT KM
:ANALYSIS:DUNIT? -> :ANALYSIS:DUNIT MILE

:ANALysis:EMARker:LMTechnique

Function Sets or queries the approximation method (for events).
Syntax :ANALysis:EMARker:LMTechnique {LSA|TPA}
:ANALysis:EMARker:LMTechnique?
LSA: Least-square method
TPA: Two-point method
Example :ANALYSIS:EMARKER:LMTECHNIQUE LSA
:ANALYSIS:EMARKER:LMTECHNIQUE?
-> :ANALYSIS:EMARKER:LMTECHNIQUE TPA

:ANALysis:EMARker:SET:M1

Function Sets or queries marker M1 of the current event.
Syntax :ANALysis:EMARker:SET:M1
:ANALysis:EMARker:SET:M1?
Example :ANALYSIS:EMARKER:SET:M1
:ANALYSIS:EMARKER:SET:M1?
-> :ANALYSIS:EMARKER:SET:M1
5.0471900E+03
Description The marker is set to the cursor position.

:ANALysis:EMARker:SET:M2

Function Sets or queries marker M2 of the current event.
Syntax :ANALysis:EMARker:SET:M2
:ANALysis:EMARker:SET:M2?
Example :ANALYSIS:EMARKER:SET:M2
:ANALYSIS:EMARKER:SET:M2?
-> :ANALYSIS:EMARKER:SET:M2
7.2463500E+03
Description The marker is set to the cursor position.

:ANALysis:EMARker:SET:M3

Function Sets or queries marker M3 of the current event.

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

Example
```
:ANALYSIS:EMARKER:SET:M3
:ANALYSIS:EMARKER:SET:M3?
-> :ANALYSIS:EMARKER:SET:M3
9.0471600E+03
```

Description The marker is set to the cursor position.

:ANALysis:EMARker:SET:Y2

Function Sets or queries marker Y2 of the current event.

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

Example
```
:ANALYSIS:EMARKER:SET:Y2
:ANALYSIS:EMARKER:SET:Y2?
-> :ANALYSIS:EMARKER:SET:Y2
7.3777700E+03
```

Description The marker is set to the cursor position.

:ANALysis:EVENt:CURRent:CUMLoss?

Function Sets or queries the cumulative loss of the current event.

Syntax
```
:ANALysis:EVENt:CURRent:CUMLoss?
```

Example
```
:ANALYSIS:EVENT:CURRENT:CUMLOSS?
-> 1.810
```

Description Use this command after you use the :ANALysis:EVENt:CURRent:INDex command to set an index number.

:ANALysis:EVENt:CURRent:DISTance?

Function Acquires the distance of the current event.

Syntax
```
:ANALysis:EVENt:CURRent:DISTance?
```

Example
```
:ANALYSIS:EVENT:CURRENT:DISTANCE?
-> 987.000
```

Description Use this command after you use the :ANALysis:EVENt:CURRent:INDex command to set an index number.

ANALysis:EVENt:CURRent:INDex

Function Moves the current event.

Syntax
```
:ANALysis:EVENt:CURRent:INDex {<NRf>}
```
<NRf> = Event number
1 to 100: Event number
-1: S point, R point
0: E point

Example
```
:ANALYSIS:EVENT:CURRENT:INDEX 5
```

Description The event that corresponds to the specified number becomes the current event.

:ANALysis:EVENt:CURRent:LOSS?

Function Acquires the connection loss of the current event.

Syntax
```
:ANALysis:EVENt:CURRent:LOSS?
```

Example
```
:ANALYSIS:EVENT:CURRENT:LOSS?
-> 2.000
```

Description Use this command after you use the :ANALysis:EVENt:CURRent:INDex command to set an index number.

:ANALysis:EVENt:CURRent:RETurnloss?

Function Acquires the optical return loss of the current event.

Syntax
```
:ANALysis:EVENt:CURRent:RETurnloss?
```

Example
```
:ANALYSIS:EVENT:CURRENT:RETURNLOSS?
-> 2.000
```

Description Use this command after you use the :ANALysis:EVENt:CURRent:INDex command to set an index number.

:ANALysis:EVENt:CURRent:TYPE?

Function Acquires the event type of the current event.

Syntax
```
:ANALysis:EVENt:CURRent:TYPE?
```

Example
```
:ANALYSIS:EVENT:CURRENT:TYPE?
-> REFLECTANCE
```
REFLectance: Reflectance
SPLus: Positive loss
SMINus: Negative loss

Description Use this command after you use the :ANALysis:EVENt:CURRent:INDex command to set an index number.

:ANALysis:EVENt:CURRent:UNITloss?

Function Acquires the dB/km of the current event.

Syntax
```
:ANALysis:EVENt:CURRent:UNITloss?
```

Example
```
:ANALYSIS:EVENT:CURRENT:UNITLOSS?
-> 2.000
```

Description Use this command after you use the :ANALysis:EVENt:CURRent:INDex command to set an index number.

:ANALysis:EVENt:DELete
Function Deletes the current event.
Syntax :ANALysis:EVENt:DELete
Example :ANALYSIS:EVENT:DELETE

:ANALysis:EVENt:FIX:MODE
Function Sets or queries the event fix mode.
Syntax :ANALysis:EVENt:FIX:MODE <MODE1|MODE2>
:ANALysis:EVENt:FIX:MODE?
Example :ANALysis:EVENt:FIX:MODE MODE1
:ANALysis:EVENt:FIX:MODE?
-> :ANALysis:EVENt:FIX:MODE MODE1

:ANALysis:EVENt:FIX:STATe
Function Sets or queries the event fix on/off state.
Syntax :ANALysis:EVENt:FIX:STATe <Boolean>
:ANALysis:EVENt:FIX:STATe?
Example :ANALysis:EVENt:FIX:STATe ON
:ANALysis:EVENt:FIX:STATe?
-> :ANALysis:EVENt:FIX:STATe 1

:ANALysis:EVENt:INSert
Function Inserts an event at the cursor position.
Syntax :ANALysis:EVENt:INSert
Example :ANALYSIS:EVENT:INSERT

:ANALysis:EVENt:IOR
Function Sets or queries the section group index of the current event.
Syntax :ANALysis:EVENt:IOR {<NRf>}
:ANALysis:EVENt:IOR?
<NRf> = 1.30000 to 1.79999 (in steps of 0.00001)
Example :ANALYSIS:EVENT:IOR 1.47
:ANALYSIS:EVENT:IOR?
-> :ANALYSIS:EVENT:IOR 1.46000

:ANALysis:FEDetection
Function Sets or queries whether or not fault events are detected.
Syntax :ANALysis:FEDetection {<Boolean>}
:ANALysis:FEDetection?
Example :ANALYSIS:FEDETECTION ON
:ANALYSIS:FEDETECTION?
-> :ANALYSIS:FEDETECTION 0

:ANALysis:FMARker:DELete
Function Deletes the marker.
Syntax :ANALysis:FMARker:DELete
Example :ANALYSIS:FMARKER:DELETE

:ANALysis:FMARker:LEFT:DISTance?
Function Queries the distance between markers 1 and 2.
Syntax :ANALysis:FMARker:LEFT:DISTance?
Example :ANALYSIS:FMARKER:LEFT:DISTANCE?
-> :ANALYSIS:FMARKER:LEFT:DISTANCE 490.20000E+00

:ANALysis:FMARker:LEFT:LOSS?
Function Queries the loss between markers 1 and 2.
Syntax :ANALysis:FMARker:LEFT:LOSS?
Example :ANALysis:FMARker:LEFT:LOSS?
-> :ANALYSIS:FMARKER:LEFT:LOSS 137.00000E-03

:ANALysis:FMARker:LEFT:UNITloss?
Function Queries the slope between markers 1 and 2.
Syntax :ANALysis:FMARker:LEFT:UNITloss?
Example :ANALYSIS:FMARKER:LEFT:UNITLOSS?
-> :ANALYSIS:FMARKER:LEFT:UNITLOSS 279.00000E-03

:ANALysis:FMARker:LMTechnique
Function Sets or queries the approximation method (for markers).
Syntax :ANALysis:FMARker:LMTechnique {LSA|TPA}
:ANALysis:FMARker:LMTechnique?
LSA: Least-square method
TPA: Two-point method
Example :ANALYSIS:FMARKER:LMTECHNIQUE LSA
:ANALYSIS:FMARKER:LMTECHNIQUE?
-> :ANALYSIS:FMARKER:LMTECHNIQUE TPA

:ANALysis:FMARker:LOSS?
Function Queries the connection loss.
Syntax :ANALysis:FMARker:LOSS?
Example :ANALYSIS:FMARKER:LOSS?
-> :ANALYSIS:FMARKER:LOSS 162.00000E-03

:ANALysis:FMARker:REFLection:SATurated?
Function Queries the reflection saturation.
Syntax :ANALysis:FMARker:REFLection:SATurated?
0: Not saturated
1: Saturated
Example :ANALYSIS:FMARKER:REFLECTION:SATURATED?
-> :ANALYSIS:FMARKER:REFLECTION:SATURATED 0

:ANALysis:FMARker:REFLection:VALue?
Function Queries the amount of reflection.
Syntax :ANALysis:FMARker:REFLection:VALue?
Example :ANALYSIS:FMARKER:REFLECTION:VALUE?
-> :ANALYSIS:FMARKER:REFLECTION:VALUE 1.0640000E+00

:ANALysis:FMARker:RETurnloss:SATurated?

Function Queries the optical return loss saturation.

Syntax
```
:ANALysis:FMARker:RETurnloss:
SATurated?
```
0: Not saturated

1: Saturated

Example
```
:ANALYSIS:FMARKER:RETURNLOSS:
SATURATED?
-> :ANALYSIS:FMARKER:RETURNLOSS:
SATURATED 1
```

:ANALysis:FMARker:RETurnloss:VALue?

Function Queries the optical return loss.

Syntax
```
:ANALysis:FMARker:RETurnloss:VALue?
```
Example
```
:ANALYSIS:FMARKER:RETURNLOSS:VALUE?
-> :ANALYSIS:FMARKER:RETURNLOSS:VALUE
47.003000E+00
```

:ANALysis:FMARker:RIGHt:DISTance?

Function Queries the distance between markers 2 and 3.

Syntax
```
:ANALysis:FMARker:RIGHt:DISTance?
```
Example
```
:ANALYSIS:FMARKER:RIGHT:DISTANCE?
-> :ANALYSIS:FMARKER:RIGHT:DISTANCE
232.95000E+00
```

:ANALysis:FMARker:RIGHt:LOSS?

Function Queries the loss between markers 2 and 3.

Syntax
```
:ANALysis:FMARker:RIGHt:LOSS?
```
Example
```
:ANALYSIS:FMARKER:RIGHT:LOSS?
-> :ANALYSIS:FMARKER:RIGHT:LOSS
-159.00000E-03
```

:ANALysis:FMARker:RIGHt:UNITloss?

Function Queries the slope between markers 2 and 3.

Syntax
```
:ANALysis:FMARker:RIGHt:UNITloss?
```
Example
```
:ANALYSIS:FMARKER:RIGHT:UNITLOSS?
-> :ANALYSIS:FMARKER:RIGHT:UNITLOSS
-683.00000E-03
```

:ANALysis:FMARker:SET:M<x>

Function Sets or queries markers.

Syntax
```
:ANALysis:FMARker:SET:M<x>
:ANALysis:FMARker:SET:M<x>?
<x> = 1, 2, 3
```
Example
```
:ANALYSIS:FMARKER:SET:M1
:ANALYSIS:FMARKER:SET:M1?
-> :ANALYSIS:FMARKER:SET:M1
7.2648300E+03
```

:ANALysis:FMARker:SET:Y<x>

Function Sets or queries auxiliary markers.

Syntax
```
:ANALysis:FMARker:SET:Y<x>
:ANALysis:FMARker:SET:Y<x>?
<x> = 1, 2, 3
```
Example
```
:ANALYSIS:FMARKER:SET:Y2
:ANALYSIS:FMARKER:SET:Y2?
-> :ANALYSIS:FMARKER:SET:Y2
6.1786000E+03
```

:ANALysis:IOR

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

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

Example
```
:ANALYSIS:IOR 1.48000
:ANALYSIS:IOR?
-> :ANALYSIS:IOR 1.48000
```

:ANALysis:MACRobending:DISPlay

Function Sets or queries the macrobending display on/off state.

Syntax
```
:ANALysis:MACRobending:DISPlay
{<Boolean>}
```
Example
```
:ANALYSIS:MACROBENDING:DISPLAY ON
:ANALYSIS:MACROBENDING:DISPLAY?
> :ANALYSIS:MACROBENDING:DISPLAY 1
```

:ANALysis:MACRobending:THReshold

Function Sets or queries the macrobending threshold.

Syntax
```
:ANALysis:MACRobending:THReshold
{<NRf>}
<NRf>: 0.001 to 99.999 [dB]
```
Example
```
:ANALYSIS:MACROBENDING:THRESHOLD 1.000
:ANALYSIS:MACROBENDING:THRESHOLD?
-> :ANALYSIS:MACROBENDING:THRESHOLD
```

: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 its current position.

Syntax
```
:ANALysis:REFerence:DISTance
:ANALysis:REFerence:DISTance?
```
Example
```
:ANALYSIS:REFERENCE:DISTANCE
:ANALYSIS:REFERENCE:DISTANCE?
-> :ANALYSIS:REFERENCE:DISTANCE
39.999710E+03
```
Description A query returns a value in meters based on the mouth of the OTDR.

:ANALysis:THReshold:EOFiber

Function Sets or queries the fiber end threshold.

Syntax
```
:ANALysis:THReshold:EOFiber {<NRf>}
:ANALysis:THReshold:EOFiber?
```
<NRf> = 3 to 65

Example
```
:ANALYSIS:THRESHOLD:EOFIBER 5
:ANALYSIS:THRESHOLD:EOFIBER?
-> :ANALYSIS:THRESHOLD:EOFIBER 10
```

Description After setting the value, wait until the AS flag in the return value of the STATus:CONDition? command becomes zero, and then query the event results.

:ANALysis:THReshold:FERLoss

Function Sets or queries the fault event optical return loss threshold.

Syntax
```
:ANALysis:THReshold:FERLoss {<NRf>}
:ANALysis:THReshold:FERLoss?
```
<NRf> = 20 to 70

Example
```
:ANALYSIS:THRESHOLD:FERLOSS 50
:ANALYSIS:THRESHOLD:FERLOSS?
-> :ANALYSIS:THRESHOLD:FERLOSS 30
```

Description After setting the value, wait until the AS flag in the return value of the STATus:CONDition? command becomes zero, and then query the event results.

:ANALysis:THReshold:FESLoss

Function Sets or queries the fault event connection loss threshold.

Syntax
```
:ANALysis:THReshold:FESLoss {<NRf>}
:ANALysis:THReshold:FESLoss?
```
<NRf> = 0.01 to 9.99

Example
```
:ANALYSIS:THRESHOLD:FESLOSS 0.01
:ANALYSIS:THRESHOLD:FESLOSS?
-> :ANALYSIS:THRESHOLD:FESLOSS 1.23
```

Description After setting the value, wait until the AS flag in the return value of the STATus:CONDition? command becomes zero, and then query the event results.

:ANALysis:THReshold:RLOSs

Function Sets or queries the optical return loss threshold.

Syntax
```
:ANALysis:THReshold:RLOSs {<NRf>}
:ANALysis:THReshold:RLOSs?
```
<NRf> = 20 to 70

Example
```
:ANALYSIS:THRESHOLD:RLOSS 50
:ANALYSIS:THRESHOLD:RLOSS?
-> :ANALYSIS:THRESHOLD:RLOSS 30
```

Description After setting the value, wait until the AS flag in the return value of the STATus:CONDition? command becomes zero, and then query the event results.

:ANALysis:THReshold:SLOSs

Function Sets or queries the connection loss threshold.

Syntax
```
:ANALysis:THReshold:SLOSs {<NRf>}
:ANALysis:THReshold:SLOSs?
```
<NRf> = 0.01 to 9.99

Example
```
:ANALYSIS:THRESHOLD:SLOSS 0.01
:ANALYSIS:THRESHOLD:SLOSS?
-> :ANALYSIS:THRESHOLD:SLOSS 1.23
```

Description After setting the value, wait until the AS flag in the return value of the STATus:CONDition? command becomes zero, and then query the event results.

:ANALysis:SECTion:BASelevel?

Function Queries the dB value of the interval data reference point.

Syntax
```
:ANALysis:SECTion:BASelevel?
```

Example
```
:ANALYSIS:SECTION:BASELEVEL?
-> :ANALYSIS:SECTION:BASELEVEL
34.268000E+00
```

:ANALysis:SECTion:DELete

Function Deletes the interval analysis data.

Syntax
```
:ANALysis:SECTion:DELete
```

Example
```
:ANALYSIS:SECTION:DELETE
```

:ANALysis:SECTion:DISTance?

Function Queries the distance of the interval data.

Syntax
```
:ANALysis:SECTion:DISTance?
```

Example
```
:ANALYSIS:SECTION:DISTANCE?
-> :ANALYSIS:SECTION:DISTANCE
80.262240E+03
```

:ANALysis:SECTion:END

Function Sets or queries the end position of the interval data.

Syntax
```
:ANALysis:SECTion:END {<NRf>}
```
<NRf>: End position
```
:ANALysis:SECTion:END?
```

Example
```
:ANALYSIS:SECTION:END
:ANALYSIS:SECTION:END?
-> :ANALYSIS:SECTION:END 119.99912E+03
```

:ANALysis:SECTion:LMTechnique

Function Sets or queries the interval analysis approximation method.

Syntax
```
:ANALysis:SECTion:LMTechnique {LSA|TPA}
:ANALysis:SECTion:LMTechnique?
```
LSA: Least-square method
TPA: Two-point method

Example
```
:ANALYSIS:SECTION:LMTECHNIQUE LSA
:ANALYSIS:SECTION:LMTECHNIQUE?
-> :ANALYSIS:SECTION:LMTECHNIQUE LSA
```

:ANALysis:SECTion:LOSS?

Function Queries the loss of the interval data.

Syntax `:ANALysis:SECTion:LOSS?`

Example `:ANALYSIS:SECTION:LOSS?`
`-> :ANALYSIS:SECTION:LOSS 7.9300000E+00`

:ANALysis:SECTion:REFerence

Function Sets the interval data reference point.

Syntax `:ANALysis:SECTion:REFerence`

Example `:ANALYSIS:SECTION:REFERENCE`

:ANALysis:SECTion:RETurnloss:SATurated?

Function Queries the optical return loss saturation of the interval data.

Syntax `:ANALysis:SECTion:RETurnloss:SATurated?`

Example `:ANALYSIS:SECTION:RETURNLOSS:SATURATED?`
`-> :ANALYSIS:SECTION:RETURNLOSS:SATURATED 1`

:ANALysis:SECTion:RETurnloss:VALue?

Function Queries the optical return loss of the interval data.

Syntax `:ANALysis:SECTion:RETurnloss:VALue?`

Example `:ANALYSIS:SECTION:RETURNLOSS:VALUE?`
`-> :ANALYSIS:SECTION:RETURNLOSS:VALUE 48.250000E+00`

:ANALysis:SECTion:STARt

Function Sets or queries the start position of the interval data.

Syntax `:ANALysis:SECTion:STARt {<NRf>}`
`<NRf>`: Start position
`:ANALysis:SECTion:STARt?`

Example `:ANALYSIS:SECTION:START`
`:ANALYSIS:SECTION:START?`
`-> :ANALYSIS:SECTION:START 39.736870E+03`

:ANALysis:TRACefix:STATe

Function Sets or queries the snapshot (waveform freeze) feature.

Syntax `:ANALysis:TRACefix:STATe {<Boolean>}`
`:ANALysis:TRACefix:STATe?`

Example `:ANALYSIS:TRACEFIX:STATE ON`
`:ANALYSIS:TRACEFIX:STATE?`
`-> :ANALYSIS:TRACEFIX:STATE 1`

4.4 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 communication settings.

Syntax `:COMMunicate?`

Example `:COMMUNICATE? -> :COMMUNICATE:HEADER 1;VERBOSE 0`

:COMMunicate:HEADer

Function Sets or queries whether or not a header is added to the response to a query. (Example with header: PMETer:REFerence 5.00. Example without header: 5.00.)

Syntax `:COMMunicate:HEADer {<Boolean>}`
`:COMMunicate:HEADer?`

Example `:COMMUNICATE:HEADER ON`
`:COMMUNICATE:HEADER?`
`->:COMMUNICATE:HEADER 1`

:COMMunicate:VERBose

Function Sets or queries whether the response to a query is returned fully spelled out (example: PMETer:REFerence 5.00) or using abbreviation (example: PMET:REF 5.00).

Syntax `:COMMunicate:VERBose {<Boolean>}`
`:COMMunicate:VERBose?`

Example `:COMMUNICATE:VERBOSE ON`
`:COMMUNICATE:VERBOSE?`
`->:COMMUNICATE:VERBOSE 0`

4.5 DISPlay Group

The commands in this group deal with the screen display.
You can make the same settings and queries that you can by using the front panel.

:DISPlay:ALINe

Function Sets or queries the display of approximated lines.
Syntax `:DISPlay:ALINe {<Boolean>}`
`:DISPlay:ALINe?`
Example `:DISPLAY:ALINE ON`
`:DISPLAY:ALINE? -> :DISPLAY:ALINE 0`

:DISPlay:COLor

Function Sets or queries the screen colors.
Syntax `:DISPlay:COLor {COLOR|BLACKWHITE}`
`:DISPlay:COLor?`
`COLOR:` Color
`BLACKWHITE:` Black and white
Example `:DISPLAY:COLOR COLOR`
`:DISPLAY:COLOR?`
`-> :DISPLAY:COLOR COLOR`

:DISPlay:CURSor:DBValue

Function Sets or queries the cursor dB value.
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 or queries the second-cursor display.
Syntax `:DISPlay:CURSor:SECond {<Boolean>}`
`:DISPlay:CURSor:SECond?`
Example `:DISPLAY:CURSOR:SECOND ON`
`:DISPLAY:CURSOR:SECOND?`
`-> :DISPLAY:CURSOR:SECOND 1`

:DISPlay:DECibel:UPPer

Function Sets or queries the display start level.
Syntax `:DISPlay:DECibel:UPPer {<NRf>}`
`:DISPlay:DECibel:UPPer?`
`<NRf>` = `1.6` to `70 (dB)`
Example `:DISPLAY:DECIBEL:UPPER 60`
`:DISPLAY:DECIBEL:UPPER?`
`-> :DISPLAY:DECIBEL:UPPER 50.0`
Description The range that can be specified varies depending on the vertical axis zoom.

:DISPlay:DISTance:LEFT

Function Sets or queries the display start distance.
Syntax `DISPlay:DISTance:LEFT {<NRf>}`
`DISPlay:DISTance:LEFT?`
Example `DISPLAY:DISTANCE:LEFT 1000`
`DISPLAY:DISTANCE:LEFT?`
`-> :DISPLAY:DISTANCE:LEFT 498.97000E+00`
Description The range of values that you can set varies depending on the horizontal axis zoom.

:DISPlay:DIVide:DECibel

Function Sets or queries the vertical axis zoom.
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?`
`-> :DISPLAY:DIVIDE:DECIBEL 1.0`
Description The unit is decibels.

:DISPlay:DIVide:DISTance

Function Sets or queries the horizontal axis zoom.
Syntax `:DISPlay:DIVide:DISTance {<NRf>}`
`:DISPlay:DIVide:DISTance?`
`<NRf>` = `1m` to `10km`
Example `:DISPLAY:DIVIDE:DISTANCE 2000`
`:DISPLAY:DIVIDE:DISTANCE?`
`-> :DISPLAY:DIVIDE:DISTANCE 10.0E+03`
Description The range of values that you can set varies depending on the distance range.
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 to 500 m
2 k range: 2 to 200 m
1 k range: 1 to 100 m
500 m range: 1 to 50m

:DISPlay:EVENt:AZOom

Function Sets or queries the event auto zoom.
Syntax `:DISPlay:EVENt:AZOom {<Boolean>}`
`:DISPlay:EVENt:AZOom?`
Example `:DISPLAY:EVENT:AZOOM ON`
`:DISPLAY:EVENT:AZOOM?`
`-> :DISPLAY:EVENT:AZOOM 1`

:DISPlay:INITialize:SCALe

Function Initializes the display scale.

Syntax
```
:DISPlay:INITialize:SCale
```

Example
```
:DISPLAY:INITIALIZE:SCALE
```

:DISPlay:MARKer:INFormation

Function Sets or queries the display of marker information.

Syntax
```
:DISPlay:MARKer:INFomation {<Boolean>}
:DISPlay:MARKer:INFomation?
```

Example
```
:DISPLAY:MARKER:INFOMATION ON
:DISPLAY:MARKER:INFOMATION?
-> :DISPLAY:MARKER:INFOMATION 1
```

:DISPlay:SCALe

Function Sets or queries the scale display.

Syntax
```
:DISPlay:SCALe {<Boolean>}
:DISPlay:SCALe?
```

Example
```
:DISPLAY:SCALE ON
:DISPLAY:SCALE?
-> :DISPLAY:SCALE 1
```

:DISPlay:TOOLtip

Function Sets or queries the tool tip display.

Syntax
```
:DISPlay:TOOLtip {<Boolean>}
```

Example
```
:DISPLAY:TOOLTIP ON
:DISPLAY:TOOLTIP?
-> :DISPLAY:TOOLTIP 1
```

4.6 FILE Group

The commands in this group deal with operations such as saving and loading data. You can perform the same operations and make the same settings and queries that you can by using the front panel.

:FILE:DELete:EXECute

Function Deletes files.
Syntax `:FILE:DELete:EXECute {<Character string>}`
<Character string> = File or folder name
Example `:FILE:DELETE:EXECUTE "0001.SOR"`
Description You can also delete folders.

:FILE:DRIVe:FREE?

Function Queries the amount of free space on the current drive.
Syntax `:FILE:DRIVe:FREE?`
Example `:FILE:DRIVE:FREE? -> 1234567`
Description The value is the number of bytes.

:FILE:DRIVe:SET

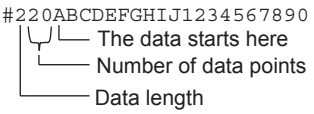
Function Sets or queries the current drive setting.
Syntax `:FILE:DRIVe:SET {INTERNAL|USB}`
`:FILE:DRIVe:SET?`
`INTERNAL: Internal memory`
`USB: USB memory`
Example `:FILE:DRIVE:SET USB`
`:FILE:DRIVE:SET?`
`-> :FILE:DRIVE:SET INTERNAL`

:FILE:FILE:EXISt?

Function Checks whether or not the specified file exists.
Syntax `:FILE:FILE:EXISt?`
Example `:FILE:FILE:EXIST?`
`-> :FILE:FILE:EXIST 1`
Description Returns 1 if the file exists and 0 if the file does not exist.

:FILE:FILE:GET?

Function Acquires the specified file.
Syntax `:FILE:FILE:GET?`
Example `:FILE:FILE:GET?`
`-> #220ABCDEFGHIJ1234567890`
Description A block data header is attached to the start of the loaded data.



:FILE:FILE:NAME

Function Specifies the file name used for file acquisition, file size acquisition, and file transfer.
Syntax `:FILE:FILE:NAME {<Character string>}`
`:FILE:FILE:NAME?`
Example `:FILE:FILE:NAME "0.SOR"`
`:FILE:FILE:NAME?`
`-> :FILE:FILE:NAME "0.sor"`
Description
- The following characters cannot be used in folder names:
", *, /, :, <, >, ?, \, |
- The following character strings cannot be used in folder names:
“AUX,” “CON,” “PRN,” “NUL,” “CLOCK,” “LPT1,” “LPT2,” “LPT3,” “LPT4,” “LPT5,” “LPT6,” “LPT7,” “LPT8,” “LPT9,” “COM1,” “COM2,” “COM3,” “COM4,” “COM5,” “COM6,” “COM7,” “COM8,” “COM9”

:FILE:FILE:SIZE?

Function Acquires the size of the specified file.
Syntax `:FILE:FILE:SIZE?`
Example `:FILE:FILE:SIZE?`
`-> :FILE:FILE:SIZE 230781`

:FILE:FOLDer:LIST?

Function Acquires a list of the contents of the current folder.
Syntax `:FILE:FOLDer:LIST?`
Example `:FILE:FOLDER:LIST? -> :FILE:FOLDER:LIST "3,ABC.SOR,DEF.SOR,MACRO/"`
Description Returns the number of results followed by the file and folder names. Folder names are followed by slashes.

:FILE:FOLDer:MAKE

Function Creates a folder.
Syntax `:FILE:FOLDer:MAKE {<Character string>}`
<Character string> = Folder name
Example `:FILE:FOLDER:MAKE "Data"`

:FILE:FOLDer:PATH

Function Sets or queries the current folder name.

Syntax
```
:FILE:FOLDer:PATH {<Character string>}
:FILE:FOLDer:PATH?
```
<Character string> = Folder name

Example
```
:FILE:FOLDER:PATH "AQ7270"
:FILE:FOLDER:PATH?
-> :FILE:FOLD:PATH "AQ7270"
```

Description
- Moves up or down the hierarchy from the current location one directory at a time (specify ".." to return to the parent directory).
- The following characters cannot be used in folder names:
  ", *, /, :, <, >, ?, \, |
- The following character strings cannot be used in folder names:"AUX," "CON," "PRN," "NUL," "CLOCK," "LPT1," "LPT2," "LPT3," "LPT4," "LPT5," "LPT6," "LPT7," "LPT8," "LPT9," "COM1," "COM2," "COM3," "COM4," "COM5," "COM6," "COM7," "COM8," "COM9"

:FILE:LOAD:EXECute

Function Loads a file.

Syntax
```
:FILE:LOAD:EXECute {<Character string>}
```
<Character string> = File name

Example
```
:FILE:LOAD:EXECUTE "0001.SOR"
```

:FILE:SAVE:COMMent

Function Sets or queries the save comment.

Syntax
```
:FILE:SAVE:COMMent {<Character string>}
:FILE:SAVE:COMMent?
```

Example
```
:FILE:SAVE:COMMENT "AQ1200"
:FILE:SAVE:COMMENT? -> "AQ1200"
```

Description You can enter up to 30 characters.

:FILE:SAVE:EXECute

Function Saves the file.

Syntax
```
:FILE:SAVE:EXECute
```

Example
```
:FILE:SAVE:EXECUTE
```

Description To check for errors after saving, first make sure that files are not being accessed by executing the STATus:CONDition? command, and then execute the STATus:ERRor command.
(See section 5.1, "Status Register.")

:FILE:SAVE:ID

Function Sets or queries the save ID.

Syntax
```
:FILE:SAVE:ID {<NRf>}
:FILE:SAVE:ID?
```
<NRf> = 0 to 9999

Example
```
:FILE:SAVE:ID 100
:FILE:SAVE:ID? -> :FILE:SAVE:ID 100
```

:FILE:SAVE:ITEM<x>

Function Sets or queries the saved items.

Syntax
```
:FILE:SAVE:ITEM<x> {NONE|WAVELENGTH|ID|
COMMENT|COMPANY|OPERATOR|CABLEID|FIBER
ID|CABLECODE|ORIGINATING|TERMINATING}
:FILE:SAVE:ITEM<x>?
```
<x> = 1 to 3

NONE:	None
WAVELENGTH:	Wavelength
ID:	ID
COMMENT:	Comment
COMPANY:	Company name
OPERATOR:	Name
CABLEID:	Cable ID
FIBERID:	Fiber ID
FIBERTYPE:	Fiber type
CABLECODE:	Cable code
ORIGINATING:	Start position
TERMINATING:	End position

Example
```
:FILE:SAVE:ITEM1 WAVELENGTH
:FILE:SAVE:ITEM1?
-> :FILE:SAVE:ITEM WAVELENGTH
```

Description You cannot specify NONE when <x> = 1.

:FILE:SAVE:SEParator

Function Sets or queries the separator used when files are saved.

Syntax
```
:FILE:SAVE:SEParator {NONE|UNDERBAR|TIL
DE|HAT}
:FILE:SAVE:SEParator?
```

NONE:	None
UNDERBAR:	Underscore (_)
TILDE:	Tilde (~)
HAT:	Hat (^)

Example
```
:FILE:SAVE:SEPARATOR UNDERBAR
:FILE:SAVE:SEPARATOR?
-> :FILE:SAVE:SEPARATOR UNDERBAR
```

:FILE:SAVE:SUB

Function Sets or queries the sub number used when files are saved.

Syntax
```
:FILE:SAVE:SUB {OFF|AB|AC|AD|AE|AF|AG
|AH}
:FILE:SAVE:SUB?
```

OFF:	OFF
AB:	a-b
AC:	a-c
AD:	a-d
AE:	a-e
AF:	a-f
AG:	a-g
AH:	a-h

Example
```
:FILE:SAVE:SUB AH
:FILE:SAVE:SUB? -> :FILE:SAVE:SUB OFF
```

:FILE:SAVE:TYPE

Function Sets or queries the filename type used when files are saved.

Syntax
```
:FILE:SAVE:TYPE {NO|COMMENT|CMNO|
NOCM|CMWLNO|NOCMWL|WLCMNO}
:FILE:SAVE:TYPE?
```
NO: Number
COMMENT: Comment
CMNO: Comment + number
NOCM: Number + comment
CMWLNO: Comment + wavelength + number
NOCMWL: Number + comment + wavelength
WLCMNO: Wavelength + comment + number

Example
```
:FILE:SAVE:TYPE COMMENT
:FILE:SAVE:TYPE?
-> :FILE:SAVE:TYPE CMWLNO
```

Description This is a simplified version of the :FILE:SAVE:ITEM<x> command. The AQ1200 returns "UNKNOWN" for items other than number (ID no.), comment, and wavelength.

:FILE:SOR:GET?

Function Acquires an SOR file image.

Syntax
```
:FILE:SOR:GET?
```

Example
```
:FILE:SOR:GET? -> #6123456ABCDEFGHJ
```

Description
- You can perform acquisition when SOR measurement data is enabled. You can use : FILE:SOR:VALid? to query whether the SOR measurement data is enabled.
- The SOR file image is acquired as block data (in binary format).

:FILE:SOR:VALid?

Function Queries whether the SOR file image is enabled.

Syntax
```
:FILE:SOR:VALID?
```

Example
```
:FILE:SOR:VALID? -> :FILE:SOR:VALID 1
```

Description Measured data
0: Disabled
1: Enabled

:FILE:SUBFolder:LIST?

Function Acquires a list of the subfolders in the current folder.

Syntax
```
:FILE:SUBFolder:LIST?
```

Example
```
:FILE:SUBFOLDER:LIST? -> :FILE:
SUBFOLDER:LIST "3,ABC/,DEF/,MACRO/"
```

Description Returns the number of results followed by the file and folder names. Folder names are followed by slashes.

:FILE:TYPE

Function Sets or queries the type of file that is saved.

Syntax
```
:FILE:TYPE {SET|SOR|CSV_WAVE|
CSV_EVENT|BMP|PNG|JPG|CFG}
:FILE:TYPE?
```
SET: A setup file
SOR: A file that conforms to Telcordia SR-4731
CSV_WAVE: A CSV waveform file
CSV_EVENT: A CSV event file
BMP: A BMP file
PNG: A PNG file
JPG: A JPG file
CFG: System setup data file

Example
```
:FILE:TYPE SOR
:FILE:TYPE? -> :FILE:TYPE CSV_WAVE
```

Description SOR, CSV_WAVE, and CSV_ENENT are only valid when the function mode is set to OTDR.

4.7 LABel Group

The commands in this group deal with the settings of the labels used in file identification. You can perform the same operations and make the same settings and queries that you can by using the front panel.

:LABel:CABLe:CODE

Function Sets or queries the cable code.

Syntax `:LABel:CABLe:CODE {<Character string>}`
`:LABel:CABLe:CODE?`
<Character string> = Up to 36 characters

Example
```
:LABEL:CABLE:CODE "123"
:LABEL:CABLE:CODE
->:LAB:CABLE:CODE "123"
```

:LABel:CABLe:ID

Function Sets or queries the cable ID.

Syntax `:LABel:CABLe:ID {<Character string>}`
`:LABel:CABLe:ID?`
<Character string> = Up to 36 characters

Example
```
:LABEL:CABLE:ID "123"
:LABEL:CABLE:ID?
->:LABEL:CABLE:ID "123"
```

:LABel:COMPany

Function Sets or queries the company name.

Syntax `:LABel:COMPany {<Character string>}`
`:LABel:COMPany?`
<Character string> = Up to 36 characters

Example
```
:LABEL:COMPANY "123"
:LABEL:COMPANY?
->:LABEL:COMPANY "123"
```

:LABel:DFLag:CURRent

Function Sets or queries the current data flag.

Syntax `:LABel:DFLag:CURRent {BC|RC|OT|CC}`
`:LABel:DFLag:CURRent?`
`BC:` as-Built Condition
`RC:` as-Repaired Condition
`OT:` Other
`CC:` Current Condition

Example
```
:LABEL:DFLAG:CURRENT BC
:LABEL:DFLAG:CURRENT?
-> :LABEL:DFLAG:CURRENT BC
```

:LABel:FIBer:ID

Function Sets or queries the fiber ID.

Syntax `:LABel:FIBer:ID {<Character string>}`
`:LABel:FIBer:ID?`
<Character string> = Up to 36 characters

Example
```
:LABEL:FIBER:ID "123"
:LABEL:FIBER:ID?
-> :LABEL:FIBER:ID "123"
```

:LABel:FIBer:TYPE

Function Sets or queries the fiber type.

Syntax `:LABel:FIBer:TYPE {SMF|DSF|NZ_DSF|MMF}`
`:LABel:FIBer:TYPE?`
`SMF:` Single-mode optical fiber
`DSF:` Dispersion-shifted single-mode fiber
`NZ_DSF:` Non-zero dispersion-shifted single-mode fiber
`MMF:` Multi-mode optical fiber

Example
```
:LABEL:FIBER:TYPE SMF
:LABEL:FIBER:TYPE?
-> :LABEL:FIBER:TYPE SMF
```

:LABel:LOCation:ORIGinating

Function Sets or queries the start position label.

Syntax `:LABel:LOCation:ORIGinating {<Character string>}`
`:LABel:LOCation:ORIGinating?`
<Character string> = Up to 36 characters

Example
```
:LABEL:LOCATION:ORIGINATING "123"
:LABEL:LOCATION:ORIGINATING?
-> :LABEL:LOC:ORIGINATING "123"
```

:LABel:LOCation:TERMinating

Function Sets or queries the end position label.

Syntax `:LABel:LOCation:TERMinating {<Character string>}`
`:LABel:LOCation:TERMinating?`
<Character string> = Up to 36 characters

Example
```
:LABEL:LOCATION:TERMINATING "123"
:LABEL:LOCATION:TERMINATING?
-> :LABEL:LOCATION:TERMINATING "123"
```

:LABel:OPERator

Function Sets or queries the operator name.

Syntax `:LABel:OPERator {<Character string>}`
`:LABel:OPERator?`
<Character string> = Up to 36 characters

Example
```
:LABEL:OPERATOR "123"
:LABEL:OPERATOR?
-> :LABEL:OPERATOR "123"
```

4.8 LIGHtsource Group

The commands in this group deal with the light source. You can perform the same operations and make the same settings and queries that you can by using the front panel. The light source can only be used on models with the loss test option (/SLT or /HLT).

:LIGHtsource:ABORt

Function Turns the measurement light off.
Syntax :LIGHtsource:ABORt
Example :LIGHTSOURCE:ABORT

:LIGHtsource:EXECute

Function Turns the measurement light on.
Syntax :LIGHtsource:EXECute
Example :LIGHTSOURCE:EXECUTE

:LIGHtsource:MODulation

Function Sets or queries the light source modulation frequency.
Syntax :LIGHtsource:MODulation {MOD_CW|MOD_270HZ|MOD_1KHZ|MODE_2KHZ}
:LIGHtsource:MODulation?
Example :LIGHTSOURCE:MODULATION MOD_CW
:LIGHTSOURCE:MODULATION?
-> :LIGHTSOURCE:MODULATION MOD_CW
Description When :LIGHtsource:WAVelength is set to 0.850 um or 1.300 um, the modulation frequency can only be set to CW or 270 Hz.

:LIGHtsource:STATe

Function Turns the measurement light off or on.
Syntax :LIGHtsource:STATe <Boolean>
Example :LIGHT:SOURCE ON
:LIGHT:SOURCE? -> LIGHT:SOURCE ON
Description OFF: The same as LIGHtsource:ABORt.
ON: The same as LIGHtsource:EXECute.

:LIGHtsource:WAVelength

Function Sets or queries the light source wavelength.
Syntax :LIGHtsource:WAVelength {<NRf>}
:LIGHtsource:WAVelength?
<NRf> = 0.850um to 1.650um
(850E-9 to 1650E-9)
Example :LIGHTSOURCE:WAVELENGTH 0.85um
:LIGHTSOURCE:WAVELENGTH 1650E-9
:LIGHTSOURCE:WAVELENGTH 1.650E-6
:LIGHTSOURCE:WAVELENGTH?
-> :LIGHTSOURCE:WAVELENGTH 1550E-09
Description The wavelengths that can be specified vary depending on the model.

4.9 MENU Group

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

:MENU:ERRor:CLEar

Function Clears the error dialog box.

Syntax `:MENU:ERRor:CLEar`

Example `:MENU:ERROR:CLEAR`

:MENU:FUNCtion

Function Sets or queries the function mode.

Syntax `:MENU:FUNCtion {TOP|OTDR|LSPM|LOSStest|MLOSstest|IPTest|VLS|FAULtlocator|FIBerscope}`

`:MENU:FUNCtion?`

Example `:MENU:FUNCTION TOP`

`:MENU:FUNCTION? -> :MENU:FUNCTION LSPM`

`TOP:`	Top menu
`OTDR:`	OTDR
`LSPM`[1]`:`	LSPM Light source power meter
`LOSStest`[1]`:`	Auto loss test
`MLOSstest`[1]`:`	Multi-core loss test
`IPtest`[2]`:`	IP test
`VLS`[3]`:`	Visible light source
`FAULtlocator:`	Fault locator
`FIBerscope:`	Fiber inspection probe

1 Can only be selected on models with the loss test option (/SLT or /HLT).

2 Can only be selected on models with the LAN option.

3 Can only be selected on models with the visible light source option.

:MENU:MARKer

Function Sets or queries the marker mode.

Syntax `:MENU:MARKer {MARKER|LINE}`

`:MENU:MARKer?`

`MARKER:`	Marker
`LINE:`	Line

Example `:MENU:MARKER MARKER`

`:MENU:MARKER?`

`-> :MENU:MARKER MARKER`

4.10 MISC Group

The commands in this group deal with the date, language, and power management. You can make the same settings and queries that you can by using the front panel.

:MISC:ALARmsound

Function Sets or queries the alarm sound.
Syntax `:MISC:ALARmsound {<Boolean>}`
`:MISC:ALARmsound?`
Example `:MISC:ALARMSOUND OFF`
`:MISC:ALARMSOUND?`
`-> :MISC:ALARMSOUND 1`

:MISC:BACKlightoff

Function Sets or queries the backlight off setting when the AQ1200 is running on battery power.
Syntax `:MISC:BACKlightoff {<Boolean>}`
Example `:MISC:BACKLIGHTOFF ON`
`:MISC:BACKLIGHTOFF?`
`-> :MISC:BACKLIGHTOFF 1`
Description The screen saver feature is not enabled during OTDR measurement.

:MISC:BRIGhtness:AC

Function Sets or queries the LCD brightness when the AC adapter is connected.
Syntax `:MISC:BRIGhtness:AC {BRIGHT|NORMAL|DARK}`
`:MISC:BRIGhtness:AC?`
Example `:MISC:BRIGHTNESS:AC NORMAL`
`:MISC:BRIGHTNESS:AC?`
`-> :MISC:BRIGHTNESS:AC NORMAL`
Description Bright: BRIGHT
Normal: NORMAL
Power save: DARK

:MISC:BRIGhtness:BATTery

Function Sets or queries the LCD brightness when the AQ1200 is running on battery power.
Syntax `:MISC:BRIGhtness:BATTery {BRIGHT|NORMAL|DARK}`
`:MISC:BRIGhtness:BATTery?`
Example `:MISC:BRIGHTNESS:BATTERY NORMAL`
`:MISC:BRIGHTNESS:BATTERY?`
`-> :MISC:BRIGHTNESS:BATTERY NORMAL`
Description The parameters are the same as those for :MISC:BRIGhtness:AC.

:MISC:BACKlightoff

Function Sets or queries the backlight off setting when the AQ1200 is running on battery power.
Syntax `:MISC:BACKlightoff {<Boolean>}`
Example `:MISC:BACKLIGHTOFF ON`
`:MISC:BACKLIGHTOFF?`
`-> :MISC:BACKLIGHTOFF 1`

:MISC:DATE:DAY

Function Sets or queries the day.
Syntax `:MISC:DATE:DAY {<NRf>}`
`<NRf>` = 1 to 31 (in steps of 1)
`:MISC:DATE:DAY?`
Example `:MISC:DATE:DAY 1`
`:MISC:DATE:DAY? -> :MISC:DATE:DAY 1`
Description This setting takes effect when :MISC:DATE:SET is executed.

:MISC:DATE:GET?

Function Queries the date and time.
Syntax `:MISC:DATE:GET?`
Example `:MISC:DATE:GET?`
`-> :MISC:DATE:GET 2009/01/31 23:59:59`

:MISC:DATE:HOUR

Function Sets or queries the hour.
Syntax `:MISC:DATE:HOUR {<NRf>}`
`<NRf>` = 0 to 23 (in steps of 1)
`:MISC:DATE:HOUR?`
Example `:MISC:DATE:HOUR 17`
`:MISC:DATE:HOUR?`
`-> :MISC:DATE:HOUR 12`
Description This setting takes effect when :MISC:DATE:SET is executed.

:MISC:DATE:MINute

Function Sets or queries the minute.
Syntax `:MISC:DATE:MINute {<NRf>}`
`<NRf>` = 0 to 59 (in steps of 1)
`:MISC:DATE:MINute?`
Example `:MISC:DATE:MINUTE 5`
`:MISC:DATE:MINUTE?`
`-> :MISC:DATE:MINUTE 59`
Description This setting takes effect when :MISC:DATE:SET is executed.

:MISC:DATE:MODE

Function Sets or queries the date display type.
Syntax `:MISC:DATE:MODE {TYPE1|TYPE2|TYPE3}`
`:MISC:DATE:MODE?`
`TYPE1:` 2009/08/29 12:16
`TYPE2:` 08/29/2009 12:16
`TYPE3:` 2009/AUG/29 12:16
Example `:MISC:DATE:MODE TYPE1`
`:MISC:DATE:MODE?`
`-> :MISC:DATE:MODE TYPE2`

:MISC:DATE:MONTh

Function Sets or queries the month.

Syntax
```
:MISC:DATE:MONTh {<NRf>}
<NRf> = 1 to 12
:MISC:DATE:MONTh?
```

Example
```
:MISC:DATE:MONTh 8
:MISC:DATE:MONTH? -> MISC:DATE:MONTH 8
```

Description This setting takes effect when :MISC:DATE:SET is executed.

:MISC:DATE:SECond

Function Sets or queries the second.

Syntax
```
:MISC:DATE:SECond {<NRf>}
<NRf> = 0 to 59 (in steps of 1)
```

Example
```
:MISC:DATE:SECOND 0
:MISC:DATE:SECOND?
-> :MISC:DATE:SECOND 0
```

Description This setting takes 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:DATE:YEAR

Function Sets or queries the year.

Syntax
```
:MISC:DATE:YEAR {<NRf>}
<NRf> = 2009 to 2035 (in steps of 1)
:MISC:DATE:YEAR?
```

Example
```
:MISC:DATE:YEAR 2009
:MISC:DATE:YEAR? -> :MISC:DATE:YEAR
2009
```

Description This setting takes effect when :MISC:DATE:SET is executed.

:MISC:LANGuage

Function Sets or queries the language.

Syntax
```
:MISC:LANGuage {JAPANESE|ENGLISH}
:MISC:LANGuage?
```

Example
```
:MISC:LANGUAGE ENGLISH
:MISC:LANGUAGE? -> :MISC:LANGUAGE
ENGLISH
```

:MISC:POWersave:AC

Function Sets or queries the power-save setting when the AC adapter is connected.

Syntax
```
:MISC:POWersave:AC
{OFF|A1MIN|A5MIN|A10MIN|A30MIN}
:MISC:POWersave:AC?
```

OFF:	Disable
A1MIN:	Auto power off after 1 minute
A5MIN:	Auto power off after 5 minutes
A10MIN:	Auto power off after 10 minutes
A30MIN:	Auto power off after 30 minutes

Example
```
:MISC:POWERSAVE:AC A1MIN
:MISC:POWERSAVE:AC? -> :MISC:POWERSAVE:
AC OFF
```

Description Auto power off cannot be performed during OTDR measurement.

:MISC:POWersave:BATTery

Function Sets or queries the power-save setting when the AQ1200 is running on battery power.

Syntax
```
:MISC:POWersave:BATTery {OFF|A1MIN|
A5MIN|A10MIN|A30MIN}
:MISC:POWersave:BATTery?
```

Example
```
:MISC:POWERSAVE:BATTERY A1MIN
:MISC:POWERSAVE:BATTERY? -> :MISC:
POWERSAVE:BATTERY OFF
```

Description The parameters are the same as those for :MISC:POWersave:AC.

:MISC:RLOSsmode

Function Sets or queries the reflection display.

Syntax
```
:MISC:RLOSsmode {NORMAL|NTT}
:MISC:MARKer?
```

NORMAL:	Optical return loss
NTT:	Reflection

Example
```
:MISC:RLOSSMODE NORMAL
:MISC:RLOSSMODE ?
-> :MISC:RLOSSMODE NORMAL
```

4.11 NETWork Group

The commands in this group can only be used on models with the /LAN option.

:NETWork:CONTrol:PASSword

Function Sets or queries the password.

Syntax :NETWork:CONTrol:PASSword <character string>

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

Description You cannot use this command if NETWork:STATe is set to OFF.
The setting specified by this command takes effect when the AQ1200 is restarted. You can restart the AQ1200 by using :SYSTem:REBoot.
You do not have to set the password when the user name is “anonymous.”

:NETWork:CONTrol:TIMeout

Function Sets or queries the timeout value.

Syntax :NETWork:CONTrol:TIMeout {<NRf>|INFinite}
:NETWork:CONTrol:TIMeout?
<NRf>: 1 to 3600

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

Description The unit is seconds.
You cannot use this command if NETWork:STATe is set to OFF.
The setting specified by this command takes effect when the AQ1200 is restarted. You can restart the AQ1200 by using :SYSTem:REBoot.

:NETWork:CONTrol:USERname

Function Sets or queries the user name.

Syntax :NETWork:CONTrol:USERname <character string>

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

Description You cannot use this command if NETWork:STATe is set to OFF.
The setting specified by this command takes effect when the AQ1200 is restarted. You can restart the AQ1200 by using :SYSTem:REBoot.

:NETWork:DHCP

Function Sets or queries the DHCP on/off state.

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

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

Description Invalid: OFF or 0
Valid: ON or 1
You cannot use this command if NETWork:STATe is set to OFF.
The setting specified by this command takes effect when the AQ1200 is restarted. You can restart the AQ1200 by using :SYSTem:REBoot.

:NETWork:GATeway

Function Sets or queries the gateway.

Syntax :NETWork:GATeway <character string>
:NETWork:GATeway?

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

Description You cannot use this command if NETWork:STATe is set to OFF.
You cannot use this command if NETWork:DHCP is set to OFF.
The setting specified by this command takes effect when the AQ1200 is restarted. You can restart the AQ1200 by using :SYSTem:REBoot.

:NETWork:IPADdress

Function Sets or queries the IP address.

Syntax :NETWork:IPADdress <character string>
:NETWork:IPADdress?

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

Description You cannot use this command if NETWork:STATe is set to OFF.
You cannot use this command if NETWork:DHCP is set to OFF.
The setting specified by this command takes effect when the AQ1200 is restarted. You can restart the AQ1200 by using :SYSTem:REBoot.

:NETWork:NETMask

Function Sets or queries the subnet mask.

Syntax `:NETWork:NETMask` <character string>
`:NETWork:NETMask?`

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

Description You cannot use this command if NETWork:STATe is set to OFF.
You cannot use this command if NETWork:DHCP is set to OFF.
The setting specified by this command takes effect when the AQ1200 is restarted. You can restart the AQ1200 by using :SYSTem:REBoot.

:NETWork:STATe

Function Sets or queries the Ethernet on/off state.

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

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

Description Invalid: OFF or 0
Valid: ON or 1
The setting specified by this command takes effect when the AQ1200 is restarted. You can restart the AQ1200 by using :SYSTem:REBoot.

4.12 PMETer Group

The commands in this group deal with the power meter. You can make the same settings and queries that you can by using the front panel. The power meter can only be used on models with the loss test option (/SLT or /HLT).

:PMETer:AVERage:TIMes

Function Sets or queries the power meter average count.
Syntax `:PMETer:AVERage:TIMes <T1|T10|T50|T100>`
`:PMETer:AVERage:TIMes?`
`T1:` 1
`T10:` 10
`T50:` 50
`T100:` 100
Example `:PMETER:AVERAGE:TIMES T1`
`:PMETER:AVERAGE:TIMES? -> :PMETER:AVERAGE:TIMES T1`

:PMETer:DREF

Function Executes Dref on the power meter.
Syntax `:PMETer:DREF`
Example `:PMETer:DREF`

:PMETer:LINK:STATe

Function Sets or queries the light source power meter setting interlock.
Syntax `:PMETer:LINK:STATe {<Boolean>}`
Example `:PMETER:LINK:STATE ON`
`:PMETER:LINK:STATE?`
`-> :PMETER:LINK:STATE 1`

:PMETer:MAXMin:STATe

Function Sets or queries the MAX and MIN display on/off state.
Syntax `:PMETer:MAXMin:STATe {<Boolean>}`
Example `:PMETER:MAXMIN:STATE ON`
`:PMETER:MAXMIN:STATE?`
`-> :PMETER:MAXMIN:STATE 1`

:PMETer:MAXMin:MAX?

Function Sets or queries the maximum value.
Syntax `:PMETer:MAXMin:MAX?`
Example `:PMETER:MAXMIN:MAX? -> :PMETER:MAXMIN:MAX? 5.00`
Description You can only make this query when :PMETer:MAXMin:STATe is set to ON.

:PMETer:MAXMin:MIN?

Function Sets or queries the minimum value.
Syntax `:PMETer:MAXMin:MIN?`
Example `:PMETER:MAXMIN:MIN? -> :PMETER:MAXMIN:MIN? -5.00`
Description You can only make this query when :PMETer:MAXMin:STATe is set to ON.

:PMETer:MEASurement:DATA?

Function Queries the power meter's measured results.
Syntax `:PMETer:MEASurement:DATA?`
Example `:PMETER:MEASUREMENT:DATA?`
`-> :PMETER:MEASUREMENT:DATA 26.56`

:PMETer:MODulation

Function Sets or queries the power meter modulation.
Syntax `:PMETer:MODulation {MOD_270HZ|MOD_CW|MOD_1KHZ|MOD_2KHZ}`
Example `:PMETER:MODULATION MOD_270HZ`
`:PMETER:MODULATION?`
`-> :PMETER:MODULATION MOD_270HZ`

:PMETer:OFFSet

Function Sets or queries the power meter offset.
Syntax `:PMETer:OFFSet {<NRf>}`
`:PMETer:OFFSet?`
`<NRf>` = `-9.900` to `9.900` (in steps of 0.001)
Example `:PMETER:OFFSET -5.000`
`:PMETER:OFFSET?`
`-> :PMETER:OFFSET -3.000`

:PMETer:REFerence

Function Sets or queries the power meter reference value.
Syntax `:PMETer:REFerence {<NRf>}`
`:PMETer:REFerence?`
`<NRf>` = `-70.00` to `5.00` (in steps of 0.001) `[dBm]`
Example `:PMETER:REFERENCE 0.00`
`:PMETER:REFERENCE?`
`-> :PMETER:REFERENCE 0.00`

:PMETer:THReshold:LOWer

Function Sets or queries the power meter lower threshold value.
Syntax `:PMETer:THReshold:LOWer {<NRf>}`
`<NRf>` = `-70.00` to `5.00` (in steps of 0.001) `[dBm]`
Example `:PMETER:THRESHOLD:LOWER -70.00`
`:PMETER:THRESHOLD:LOWER?`
`-> :PMETER:THRESHOLD:LOWER -70.00`

:PMETer:THReshold:UPPer

Function Sets or queries the power meter upper threshold value.

Syntax `:PMETer:THReshold:UPPer {<NRf>}`
`<NRf>` = `-70.00` to `5.00` (in steps of 0.001) `[dBm]`

Example `:PMETER:THRESHOLD:UPPER -70`
`:PMETER:THRESHOLD:UPPER?`
`-> :PMETER:THRESHOLD:UPPER -70`

:PMETer:UNIT

Function Sets or queries the power meter display unit.

Syntax `:PMETer:UNIT {DB|DBM|W}`
`:PMETer:UNIT ?`

Example `:PMETER:UNIT DBM`
`:PMETER:UNIT ? -> :PMETER:UNIT DB`

:PMETer:WAVelength:DETail

Function Sets or queries the wavelength when the wavelength mode is set to Detail.

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

Example `:PMETer:WAVELENGTH:DETAIL 0.85um`
`:PMETer:WAVELENGTH:DETAIL 1650E-9`
`:PMETer:WAVELENGTH:DETAIL 1.650E-6`
`:PMETer:WAVELENGTH:DETAIL?`
`-> PMETER:WAVELENGTH:DETAIL 1550E-09`

Description If the wavelength mode is set to Simple or CWDM, executing this command sets the mode to Detail.

:PMETer:ZERoset

Function Executes zero set on the power meter.

Syntax `:PMETer:ZERoset`

Example `:PMETER:ZEROSET`

4.13 PON Group

The commands in this group deal with the power meter. These commands are valid when the PON option is selected. You can make the same settings and queries that you can by using the front panel.

:PON:AVERage:TIMes

Function Sets or queries the PON power meter average count.

Syntax
```
:PON:AVERage:TIMes <T1|T10|T50|T100>
:PON:AVERage:TIMes?
```

Example
```
:PON:AVERAGE:TIMES
:PON:AVERAGE:TIMES? -> :PON:AVERAGE:
TIMES T1
```

:PON:DIRection

Function Sets or queries the PON power meter wavelength.

Syntax
```
:PON:DIRection {ONT2OLT|OLT2ONU}
```

Example
```
:PON:DIRECTION ONT2OLT
:PON:DIRECTION?
-> :PON:DIRECTION ONT2OLT
```

:PON:M1310:MEASurement:DATA?

Function Queries the measured results of the 1310 nm PON power meter.

Syntax
```
:PON:M1310:MEASurement:DATA?
```

Example
```
:PON:M1310:MEASUREMENT:DATA?
-> :PON:M1310:MEASUREMENT:DATA
```

:PON:M1310:OFFSet

Function Sets or queries the 1310 nm PON power meter offset.

Syntax
```
:PON:M1310:OFFSet {<NRf>}
<NRf> = -9.900 to 9.900 (in steps of
0.001)
```

Example
```
:PON:M1310:OFFSET -9.900
:PON:M1310:OFFSET?
-> :PON:M1310:OFFSET -9.900
```

:PON:M1310:THReshold:JUDGe?

Function Queries the threshold judgment results of the 1310 nm PON power meter.

Syntax
```
:PON:M1310:THReshold:JUDGe?
```

Example
```
:PON:M1310:THRESHOLD:JUDGE?
-> :PON:M1310:THRESHOLD:JUDGE FAIL
```

Description A result of PASS or FAIL is returned.

:PON:M1310:THReshold:LOWer

Function Sets or queries the lower threshold value of the 1310 nm PON power meter.

Syntax
```
:PON:M1310:THReshold:LOWer {<NRf>}
<NRf> = -70.00 to 5.00 (in steps of 0.001) [dBm]
```

Example
```
:PON:M1310:THRESHOLD:LOWER -70.00
:PON:M1310:THRESHOLD:LOWER?
-> :PON:M1310:THRESHOLD:LOWER -70.00
```

:PON:M1310:THReshold:UPPer

Function Sets or queries the upper threshold value of the 1310 nm PON power meter.

Syntax
```
:PON:M1310:THReshold:UPPer {<NRf>}
<NRf> = -70.00 to 5.00 (in steps of 0.001) [dBm]
```

Example
```
:PON:M1310:THRESHOLD:UPPER -70.00
:PON:M1310:THRESHOLD:UPPER?
-> :PON:M1310:THRESHOLD:UPPER -70.00
```

:PON:M1490:MEASurement:DATA?

Function Queries the measured results of the 1490 nm PON power meter.

Syntax
```
:PON:MEASurement:DATA?
```

Example
```
:PON:MEASUREMENT:DATA?
-> :PON:MEASUREMENT:DATA -26.5
```

:PON:M1490:OFFSet

Function Sets or queries the 1490 nm PON power meter offset.

Syntax
```
:PON:M1490:OFFSet {<NRf>}
<NRf> = -9.900 to 9.900 (in steps of
0.001)
```

Example
```
:PON:M1490:OFFSET -9.900
:PON:M1490:OFFSET?
-> :PON:M1490:OFFSET -9.900
```

:PON:M1490:THReshold:JUDGe?

Function Queries the threshold judgment results of the 1490 nm PON power meter.

Syntax
```
:PON:M1490:THReshold:JUDGe?
```

Example
```
:PON:M1490:THRESHOLD:JUDGE?
-> :PON:M1490:THRESHOLD:JUDGE FAIL
```

Description A result of PASS or FAIL is returned.

:PON:M1490:THReshold:LOWer

Function Sets or queries the lower threshold value of the 1490 nm PON power meter.

Syntax
```
:PON:M1490:THReshold:LOWer {<NRf>}
<NRf> = -70.00 to 5.00 (in steps of 0.001) [dBm]
```

Example
```
:PON:M1490:THRESHOLD:LOWER -70.00
:PON:M1490:THRESHOLD:LOWER?
-> :PON:M1490:THRESHOLD:LOWER -70.00
```

:PON:M1490:THReshold:UPPer

Function Sets or queries the upper threshold value of the 1490 nm PON power meter.

Syntax
```
:PON:M1490:THReshold:UPPer {<NRf>}
<NRf> = -70.00 to 5.00 (in steps of 0.001) [dBm]
```

Example
```
:PON:M1490:THRESHOLD:UPPER -70.00
:PON:M1490:THRESHOLD:UPPER?
-> :PON:M1490:THRESHOLD:UPPER -70.00
```

:PON:M1550:MEASurement:DATA?

Function Queries the measured results of the 1550 nm PON power meter.

Syntax
```
:PON:M1550:MEASurement:DATA?
```

Example
```
:PON:M1550:MEASUREMENT:DATA?
-> :PON:M1550:MEASUREMENT:DATA
```

:PON:M1550:OFFSet

Function Sets or queries the 1550 nm PON power meter offset.

Syntax
```
:PON:M1550:OFFSet {<NRf>}
<NRf> = -9.900 to 9.900 (in steps of
0.001)
```

Example
```
:PON:M1550:OFFSET -9.900
:PON:M1550:OFFSET?
-> :PON:M1550:OFFSET -9.900
```

:PON:M1550:THReshold:JUDGe?

Function Queries the threshold judgment results of the 1550 nm PON power meter.

Syntax
```
:PON:M1550:THReshold:JUDGe?
```

Example
```
:PON:M1550:THRESHOLD:JUDGE?
-> :PON:M1550:THRESHOLD:JUDGE FAIL
```

Description A result of PASS or FAIL is returned.

:PON:M1550:THReshold:LOWer

Function Sets or queries the lower threshold value of the 1550 nm PON power meter.

Syntax
```
:PON:M1550:THReshold:LOWer {<NRf>}
<NRf> = -70.00 to 5.00 (in steps of
0.001) [dBm]
```

Example
```
:PON:M1550:THRESHOLD:LOWER -70.00
:PON:M1550:THRESHOLD:LOWER?
-> :PON:M1550:THRESHOLD:LOWER -70.00
```

:PON:M1550:THReshold:UPPer

Function Sets or queries the upper threshold value of the 1550 nm PON power meter.

Syntax
```
:PON:M1550:THReshold:UPPer {<NRf>}
<NRf> = -70.00 to 5.00 (in steps of
0.001) [dBm]
```

Example
```
:PON:M1550:THRESHOLD:UPPER -70.00
:PON:M1550:THRESHOLD:UPPER?
-> :PON:M1550:THRESHOLD:UPPER -70.00
```

:PON:UNIT

Function Sets or queries the PON power meter display unit.

Syntax
```
:PON:UNIT {DBM|W}
```

Example
```
:PON:UNIT DBM
:PON:UNIT ? -> :PON:UNIT DBM
```

:PON:ZERoset

Function Executes zero set on the PON power meter.

Syntax
```
:PON:ZERoset
```

Example
```
:PON:ZEROSET
```

4.14 PRINt Group

The commands in this group deal with printing. You can make the same settings and queries that you can by using the front panel.

:PRINt:COLor

Function Sets or queries the print colors.

Syntax
```
:PRINt:COLor {COLOR|BW}
:PRINt:COLor?
```
COLOR: Screen colors
BW: Black and white

Example
```
:PRINT:COLOR COLOR
:PRINT:COLOR? -> :PRINT:COLOR BW
```

:PRINt:EXECute

Function Executes printing.

Syntax
```
:PRINt:EXECute
```

Example
```
:PRINT:EXECUTE
```

:PRINt:MAKer

Function Sets or queries the printer manufacturer.

Syntax
```
:PRINt:MAKer {HP|EPSON|SEIKO}
:PRINt:MAKer?
```
HP: HP inkjet printer
EPSON: EPSON inkjet printer
SEIKO: Seiko Instruments MPU-L465

Example
```
:PRINT:MAKER HP
:PRINT:MAKER?
-> :PRINT:MAKER EPSON
```

4.15 SETup Group

The commands in this group deal with the initialization of the settings. The AQ1200 settings can be returned to their factory defaults.

:SETup:ALLinit

Function Returns all the settings to their factory default values and formats the internal memory.

Syntax `:SETup:ALLInit`

Example `:SETup:ALLINIT`

:SETup:INITialize

Function Returns all settings to their factory defaults.

Syntax `:SETup:INITialize`

Example `:SETUP:INITIALIZE`

4.16 STATus Group

The commands in this group are used to make settings and queries related to the communication status feature. There are no front panel keys that correspond to the commands in this group.

:STATus?

Function Queries all the settings for the communication status feature.
Syntax :STATus?
Example :STATUS? -> :STATUS:QENABLE1;
QMESSAGE 1

:STATus:CONDition?

Function Queries the contents of the condition register.
Syntax :STATus:CONDition?
Example :STATUS:CONDITION? -> 16
Description For information about the condition register, see section 5.1, “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 or queries whether or not messages other than errors will be stored to the error queue (on/off).
Syntax :STATus:QENable {<Boolean>}:STATus:
QENable?
Example :STATUS:QENABLE ON
:STATUS:QENABLE? -> :STATUS:QENABLE 1

:STATus:QMESsage

Function Sets or queries whether or not message information will be attached to the response to the STATus:ERRor? query (on/off).
Syntax :STATus:QMESsage {<Boolean>}
:STATus:QMESsage?
Example :STATUS:QMESSAGE OFF
:STATUS:QMESSAGE? -> :STATUS:QMESSAGE 1

4.17 SYSTem Group

The commands in this group deal with startup. There are no front panel keys that correspond to the commands in this group.

:SYSTem:REBoot

Function Restarts the AQ1200.
Syntax :SYSTem:REBoot
Example :SYSTEM:REBOOT

:SYSTem:SHUTdown

Function Shuts down the AQ1200.
Syntax :SYSTem:SHUTdown
Example :SYSTEM:SHUTDOWN

4.18 VLS Group

The commands in this group deal with the visible light source. These commands are valid when the VLS option is selected. You can perform the same operations and make the same settings and queries that you can by using the front panel.

:VLS:ABORt

Function Turns the visible light source off.
Syntax :VLS:ABORt
Example :VLS:ABORT

:VLS:EXECute

Function Turns the visible light source on.
Syntax :VLS:EXECute
Example :VLS:EXECUTE

:VLS:STATe

Function Turns the visible light source on or off.
Syntax :VLS:STATe {<Boolean>}
:VLS:STATe?
Example :VLS:STATE ON
:VLS:STATE?
->:VLS:STATE 1
Description :VLS:STATE ON is the same as VLS:EXECUTE.
:VLS:STATE OFF is the same as VLS:ABORT.
You can use :VLS:STATe? to check whether the visible light source is on or off.

4.19 WAVedata Group

The commands in this group deal with waveform data. There are no front panel keys that correspond to the commands in this group.

:WAVedata:LENGth?

Function Queries the number of waveform data points.
Syntax :WAVedata:LENGth?
Example :WAVEDATA:LENGTH?
-> :WAVEDATA:LENGTH 25000

:WAVedata:DISPlay:SEND:ASCii?

Function Queries the data of the displayed waveform in ASCII format.
Syntax :WAVedata:DISPlay:SEND:ASCii?
Example :WAVEDATA:DISPLAY:SEND:ASCII?
-> 0.000,1.234,...
Description Queries the waveform data that is displayed on the screen in ASCII format.
Return value format: XXXXX, XXXXX, etc.
Return value example: 45000 is returned for a value of 45.000 dB.

:WAVedata:DISPlay:SEND:BINary?

Function Queries the data of the displayed waveform in binary format.
Syntax :WAVedata:DISPlay:SEND:BINary?
Example :WAVEDATA:DISPLAY:SEND:BINARY?
-> #6123456ABCDEFGHJ1234567890...
Description Queries the waveform data that is displayed on the screen in block data (in binary format).
A returned value is the product of multiplying 1000 by a decibel value that is specified to the thousandths place. One value consists of 4 bytes and starts with the most-significant byte (to the thousandths place). For example, a returned value of 1000 indicates a value of 1.000 dB.

:WAVedata:OLDType:DISPlay:SEND?

Function Queries the data of the displayed waveform in Dot4 format.
Syntax :WAVedata:OLDType:DISPlay:SEND?
Example :WAVEDATA:OLDTYPE:DISPLAY:SEND?
-> #42000...
Description Queries the waveform data that is displayed on the screen in block data (in binary format).
The returned value is an integer value that is the product of multiplying 250 by the data value. The most-significant byte and least-significant byte are produced in alternation, starting with the most-significant byte (to the thousandths place).

:WAVedata:OLDType:SEND?

Function Queries the waveform data in Dot4 format.
Syntax :WAVedata:OLDType:SEND?
Example :WAVEDATA:OLDTYPE:SEND?
-> #6123456ABCDEFGHJ1234567890...
Description Queries all the waveform data in block data (in binary format).
The returned value is an integer value that is the product of multiplying 250 by the data value. The most-significant byte and least-significant byte are produced in alternation, starting with the most-significant byte (to the thousandths place).

:WAVedata:SEND:ASCii?

Function Queries the waveform data in ASCII format.
Syntax :WAVedata:SEND:ASCii?
Example :WAVEDATA:SEND:ASCII?
-> 0.000,1.234,...
Description Queries all the waveform data in ASCII format.
Return value format: XX.XXX, XX.XXX, etc.

:WAVedata:SEND:BINary?

Function Queries the waveform data in block data (in binary format).
Syntax :WAVedata:SEND:BINary?
Example :WAVEDATA:SEND:BINARY?
-> #6123456ABCDEFGHJ1234567890...
Description Queries all the waveform data in binary format.
A returned value is the product of multiplying 1000 by a decibel value that is specified to the thousandths place. One value consists of 4 bytes and starts with the most-significant byte (to the thousandths place). For example, a returned value of 1000 indicates a value of 1.000 dB.

:WAVedata:SEND:SIZE

Function Sets or queries the number of acquired data points.
Syntax :WAVedata:SEND:SIZE {<NRf>}
:WAVedata:SEND:SIZE?
<NRf> = 1 to the number of measured points
Example :WAVEDATA:SEND:SIZE {<NRF>}
:WAVEDATA:SEND:SIZE? -> :WAVEDATA:SEND:SIZE 20000
Description Values that exceed the range are rounded.

:WAVedata:SEND:STARt

Function Sets or queries the starting distance of the waveform data.

Syntax
```
:WAVedata:SEND:STARt {<NRf>}
:WAVedata:SEND:STARt?
<NRf> = 0 to measured distance (m)
```

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

Description Values that exceed the range are rounded.

4.20 Common Commands

*CLS (Clear Status)

Function Clears all event status registers that are displayed in the status byte register summary.

Syntax `*CLS`

Example `*CLS`

Description
- Clears all queues except the output queue and all event registers except the MAV summary message.
- After this command is executed, the AQ1200 is set to OCIS (Operation Complete Command Idle State) and OQIS (Operation Complete Query Idle State).

*ESE (Standard Event Status Enable)

Function Sets or queries the standard event enable register.

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

Example `*ESE 251`
`*ESE? -> 251`

Description
- Each item whose bit is set is enabled.
- The ESE is set to its default value when:
  The power is turned on.
  Zero is set.
- The ESE does not change its value when the following commands are executed:
  *RST
  *CLS
  Device clear (DCL, SDC)
- The default value is zero.

*ESR? (Standard Event Status Register)

Function Queries and clears the standard event status register.

Syntax `*ESR?`

Example `*ESR? -> 251`

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

*IDN? (Identification)

Function Queries the instrument type and firmware version.

Syntax `*IDN?`

Example `*IDN? -> YOKOGAWA,AQ1200,SN123456789, F1.00`

Description Four data fields delimited by commas are returned.

Field 1: Manufacturer	“YOKOGAWA”
Field 2: Model	“AQ1200A”
Field 3: Instrument serial number	“SN123456789”
Field 4: Firmware version	“F1.01”

*OPT? (Option)

Function Queries the option information.

Syntax `*OPT?`

Example `*OPT? -> PM,LS,LAN`

`SLT:`	Loss Test
`HLT:`	High Power Loss Test
`VLS:`	Visible light source
`LAN:`	LAN
`SB:`	Shoulder Belt

Description The installed options are returned delimited by commas.

*RST (Reset)

Function Returns all the settings except the communication settings to their factory default values.

Syntax `*RST`

Example `*RST`

Description
- This command stops the operation in progress and resets the AQ1200 to its factory defaults.
- The following items do not change:
  Output queue
  SRE
  ESE
  Calibration data that affects the AQ1200 specifications

*SRE (Service Request Enable)

Function Sets or queries the service request enable register.

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

Example
```
*SRE  250
*SRE? -> 250
```

Description
- Each item whose bit is set is enabled.
- The SRE is set to its default value when:
  The power is turned on.
  Zero is set.
- The SRE does not change its value when the following commands are executed:
  *RST
  *CLS
  Device clear (DCL, SDC)
- The default value is zero.

*STB? (Read Status Byte)

Function Queries the current status byte register value.

Syntax
```
*STB?
```

Example
```
*STB? -> 251
```

Description The STB is not cleared even if it is read.

*TST? (Self Test)

Function Executes a self-test.

Syntax
```
*TST?
```

Example
```
*TST? -> 0
```

Description The AQ1200 returns 0 if the self-test is successful and 1 if it is not.

5.1 Condition Register

The condition register indicates the internal condition of the AQ1200.

Condition register
`:STATus:CONDition?`

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	PRT	LS	PME	PMZ	PMM	0	AR	PC	FIA	AS	FILE	AVE	REAL

The meanings of each bit of the condition register are as follows:

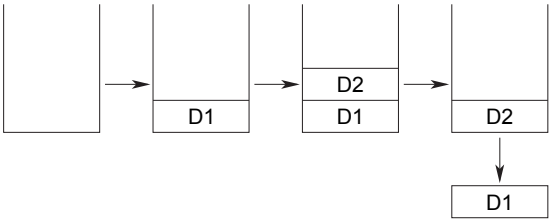
Bit 0	REAL	Set to 1 during real-time measurement.
Bit 1	AVE	Set to 1 during averaged measurement.
Bit 2	FILE	Set to 1 while a file is being accessed.
Bit 3	AS	Set to 1 during auto searching.
Bit 4	FIA	Set to 1 while the AQ1200 checks whether the fiber is in use.
Bit 5	PC	Set to 1 during plug checking.
Bit 6	AR	Set to 1 during auto range measurement.
Bit 8	PMM	Set to 1 during power meter measurement.
Bit 9	PMZ	Set to 1 while the power meter is being reset to 0.
Bit 10	PME	Set to 1 while a measurement error is occurring in the power meter.
Bit 11	LS	Set to 1 while the light source is ON.
Bit 12	PRT	Set to 1 during printing.

5.2 Output and Error Queue

Output Queue

The output queue stores query response messages. As shown below, data is stored in order and read from the oldest message first. The output queue is cleared when the AQ1200 is turned off and then back on.

The output queue cannot be emptied using the *CLS command. To see whether the output queue is emptyor not, check bit 4 (MAV) of the status byte.



Error Queue

When an error occurs, the error queue stores the error number and message. For example, if the AQ1200 receives an incorrect program message from the controller, the error number (113) and the error message (“Undefined header”) are stored in the error queue when the AQ1200 displays the error message.

You can use the `:STATus:ERRor?` query to read the contents of the error queue. Like the output queue, the messages in the error queue are read from the oldest one first.

If the error queue overflows, the last message is replaced with the following message: 350, “Queue overflow”

The error queue is cleared when:

- A `*CLS` command is received.
- The AQ1200 is turned off and then back on.

You can determine whether or not the error queue is empty by checking bit 2 in the status byte (EAV).

Appendix Error Messages

This section explains communication error messages.

- Messages can be displayed in English or in another language on the AQ1200. However, when they are read from a PC or other similar device, messages are displayed in English.
- If servicing is necessary to solve the problem indicated by a message, contact your nearest YOKOGAWA dealer.
- Only communication error messages are listed here. For other error messages, see the *User's Manual IM AQ1200-01EN*.

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.
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.

Index

Symbols

A

B

C

D

E

F

G

H

I

L

M

N

O

P

Q

R

S

T

U

V

W

Z