User's Manual

AQ6374E Optical Spectrum Analyzer Remote Control

YOKOGAWA
Yokogawa Test & Measurement Corporation

IM AQ6374E-17EN
2nd Edition

Thank you for purchasing the AQ6374E Optical Spectrum Analyzer. This remote control user's manual covers the AQ6374E. It describes the following and.

- GP-IB Interface
- Ethernet Interface
- Communication Commands
- Program Functions

To ensure correct use, please read this manual thoroughly before beginning operation. After reading this manual, keep it in a convenient location for quick reference in the event a question arises during operation. In addition to this manual, There are four manuals for the AQ6374E including this one. Read them along with this manual.

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 the instrument's performance and functions. Display contents illustrated in this manual may differ slightly from what actually appears on your screen.
- Every effort has been made in the preparation of this manual to ensure the accuracy of its contents. However, should you have any questions or find any errors, please contact your nearest YOKOGAWA dealer.
- Copying or reproducing all or any part of the contents of this manual without the permission of YOKOGAWA is strictly prohibited.

Trademarks

- Microsoft and Windows are registered trademarks or trademarks of Microsoft Corporation in the United States and/or other countries.
- Adobe and Acrobat are registered trademarks or trademarks of Adobe Systems incorporated.
- The company and product names used in this manual are not accompanied by the registered trademark or trademark symbols (® and TM).
- Other company and product names are registered trademarks or trademarks of their respective companies.

Revisions

- 1st Edition: October 2023
- 2nd Edition: February 2024

2nd Edition: February 2024 (YMI)

Manuals

The following manuals, including this one, are provided as manuals for this instrument. Please read all manuals.

Manuals Included with the Product

Manual Title	Manual No.	Description
AQ6374E Optical Spectrum Analyzer Getting Started Guide	IM AQ6374E-02EN	This guide explains the handling precautions, installation procedure, basic operations, and specifications of this instrument.
AQ6374E Optical Spectrum Analyzer Request to Download Manuals	IM AQ6374E-73EN	Describes the manuals provided on the website.
Optical Spectrum Analyzer	IM AQ6360-92Z1	A document for China.
Safety Instruction Manual	IM 00C01C01-01Z1	A document for the EU.

Manuals Provided on the Website

Download the following manuals from our website.

Manual Title	Manual No.	Description
AQ6374E Optical Spectrum Analyzer User's Manual	IM AQ6374E-01EN	Explains all functions and operating procedures of the AQ6374E except remote control and program functions.
AQ6374E Optical Spectrum Analyzer Remote Control User's Manual	IM AQ6374E-17EN	This manual. Explains functions for controlling the instrument with communication commands and program functions.

For details on downloading manuals, see Request to Download AQ6374E Manuals (IM AQ6374E-73EN).

The "EN", and "Z1" in the manual numbers are the language codes.

Online Help

The content similar to the IM AQ6374E-01EN, is included in this instrument as a help file (some the content may be omitted). For instructions on how to use the help file, see section 9.7 in the IM AQ6374E-01EN.

How to Use This Manual

Structure of the Manual

This manual contains six chapters and an appendix.

Chapter 1 Remote Control Functions

This section describes the various types of communication interfaces.

Chapter 2 GP-IB Interface

Describes the functions and lists the specifications of the GP-IB port.

Chapter 3 Ethernet Interface

Describes the functions and lists the specifications of the Ethernet interface.

Chapter 4 Programming Overview

Describes command syntax and other programming information.

Chapter 5 Commands

Describes every command individually.

Chapter 6 Status Registers

Explains the status byte and describes the various kinds of registers, cues, and other items.

Chapter 7 Program Function

Describes a program function that sets various measurement conditions and executes automated measurement, including , measurement, analysis, and data storage, without the use of an external controller.

Appendix

Provides a table of commands for the setting menu and analysis parameters for each function and explains the compatibility of commands and programs with older models such as the AQ6317/AQ6370D.

Index

An alphabetical index.

Sympols and Notation Used in This Manual

Prefixes k and K

Prefixes k and K used before units are distinguished as follows:

k ····· Denotes 1000. Example: 12 kg, 100 kHz

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

Displayed Characters

Bold characters in procedural explanations indicate panel keys and on-screen menu items that are used in the procedure.

Notes and Cautions

The notes and cautions in this manual are categorized using the following symbols.

WARNING Calls attention to actions or conditions that could cause serious or fatal injury to the user, and precautions that can be taken to prevent such occurrences.

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

French

AVERTISSEMENT Attire l'attention sur des gestes ou des conditions susceptibles de provoquer des blessures graves (voire mortelles), et sur les précautions de sécurité pouvant prévenir de tels accidents.

ATTENTION Attire l'attention sur des gestes ou des conditions susceptibles de provoquer des blessures légères ou d'endommager l'instrument ou les données de l'utilisateur, et sur les précautions de sécurité susceptibles de prévenir de tels accidents.

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

Contents

1.1 Remote Interfaces

This instrument is equipped with the following remote interfaces.

GP-IB (IEEE 488.2, See Chapter 2)

This port is used to connect a controller such as a PC to remote control this instrument. Connect a controller or another device controlled by the controller to this port.

This instrument is controlled using remote commands.

Two types of remote commands are provided: the instrument's native commands complying with SCPI (Standard Commands for Programmable Instruments), and commands compatible with the conventional model AQ6317 (see the appendix).

Ethernet (See Chapter 3)

This port is used to connect a controller such as a PC to control the instrument remotely via network.

1.2 Switching between Local and Remote

Switching from Local to Remote

When in Local mode, if a listen address is sent from the controller that sets REN (remote enable) and ATN to “True,” the instrument enters Remote mode.

- When in Remote mode, the REMOTE indicator lights.
- Keys other than the LOCAL key are disabled.
- Settings entered in Local mode are held even if switching to Remote mode.
- When an LLO (Local Lock Out) message is received from the controller, the instrument enters local lockout status. In LLO status, the LOCAL key is disabled and does not return the instrument to Local mode even when pressed. After cancelling the local lockout status, press the LOCAL key. To cancel the local lockout status, set REN to “False” from the controller.

Switching from Remote to Local

If you press the LOCAL key when in Remote mode the instrument enters Local mode. However, it does not return to Local mode if in the local lockout state.

- The REMOTE indicator turns off.
- All keys are enabled.
- Settings entered in Remote mode are held even if switching to Local mode.
- When a GTL (Go to Local) message is received from the controller, the instrument enters Local mode even if REN is set to False.

1.3 Sending/Receiving Remote Commands

Buffers

Input Buffer

The instrument's input buffer is a single stage 4 MB buffer. When receiving data that exceeds the buffer size, the data after the first megabyte is discarded. The remote command after the last command separator of the 4 MB of data is deleted.

Output Buffer

The instrument's output buffer is a single stage 4 MB buffer. Only the most recent data is held. (When a talker command is received while there is data in the buffer, the old data in the buffer is replaced with the incoming data.) When talker commands are combined and executed resulting in generation of talker data that exceeds the buffer size, the following process is carried out.

- The query error bit (QYE) of the standard event status register is set to 1.
- The talker output buffer is cleared.
- Commands received even after the buffer overflow are processed. Note, however, that talker data by talker commands is not stored at the output buffer.

Error Buffer

This instrument's error buffer is of a single stage and stores only the latest error information.

Message Terminators

This instrument allows the following message terminators to be used.

Program Message Terminators

- Assertion of EOI (End-Of-Identify) signal
- LF (line feed) character
- LF+EOI

Here, LF is a line feed (0Ah) in ASCII. For CR + LF, because CR (0Dh) is recognized as "wsp," CR + LF can consequently also be used as a message terminator. Also, for waveform binary transfer, only EOI is used as a message terminator.

Response Message Terminator

LF+EOI is used as the response message terminator.

Receiving Remote Commands

- When completing receipt of a remote command, the instrument releases the GP-IB bus.
- When receiving the next command while a command action is being executed, the instrument captures that command to store it in the receive buffer, and then releases the GP-IB bus.
- When there is a remote command in the receive buffer, the instrument does not capture a successive command even if there are commands on the GP-IB bus.
- When the action of the preceding command is complete, the instrument executes the command stored in the receive buffer and clears the buffer. Then it captures the next command into the receive buffer if there is one on the bus.
- When an output statement contains multiple remote commands, this instrument captures them all and services them in the order they were written. In this case, unless the last command in the statement has started to be executed, this instrument cannot capture the next command.

Data Inquiry

- Inquiry of data by the external controller is made using a query command or a data output request from the controller.
- Query commands end with a question mark (?).
- For query commands with an argument, the argument is specified in the form of <wsp> + <argument> at the end of the “?”.
- When a query command is received, the instrument prepares a reply to the query command in the output buffer.
- Data in the output buffer will be retained until the instrument receives an input statement or a new query command from the controller.
- If multiple query commands are specified and written in succession using a semicolon “;”, the instrument prepares replies to all of them in the output buffer. In this case, the instrument will collectively output all of the prepared data when receiving the next data output request.

Setting the timeout time

A timeout time setting of 30 seconds or more is recommended.
This instrument performs auto offset for about 30 seconds every about 10 minutes. The communication timeout of the external controller should be set to 30 seconds or more so that a timeout does not occur during the execution of the offset. See the user’s manual of your remote interface card for instructions on how to set the communication timeout time.

By default, the auto offset function is turned on in this instrument. The instrument performs offset adjustment of the analog circuits every about 10 minutes. This process takes about 30 seconds.

If you do not want to set the communication timeout to 30 seconds or less

To avoid remote malfunctions due to communication timeouts, offset processing can be performed manually. Turn off the auto offset function in advance, and then execute this function manually between measurement sequences.

Wait approximately 30 seconds until the offset process is finished. After the offset is complete, restart the measurement sequence.

The remote commands are as follows.

- Turn OFF the auto offset function
  :CALibration:ZERO off
- Perform a manual offset
  :CALibration:ZERO once

Note

- An offset interval of 10 minutes is recommended.
- If the AUTO OFFSET is OFF, the offset can fluctuate over time, and the level axis performance can degrade. Always have it turned ON.
- When the AUTO OFFSET is set to ON, AUT OFS is displayed at the bottom of the screen.

Device Trigger Function

When GET (Group Execute Trigger) is received, the instrument will perform a single sweep.

2.1 Connecting via GP-IB

GP-IB Cable

This instrument is equipped with an IEEE standard 488-1978 24-pin GP-IB connector. Use a GP-IB cable that conforms to the IEEE standard 488-1978.

Connections

Can be connected to a PC for remote control of the instrument from the PC.
Turn OFF all the power switches of the AQ6374E and any devices to be connected to it. Connect a cable to the GP-IB port on the rear panel of the instrument.

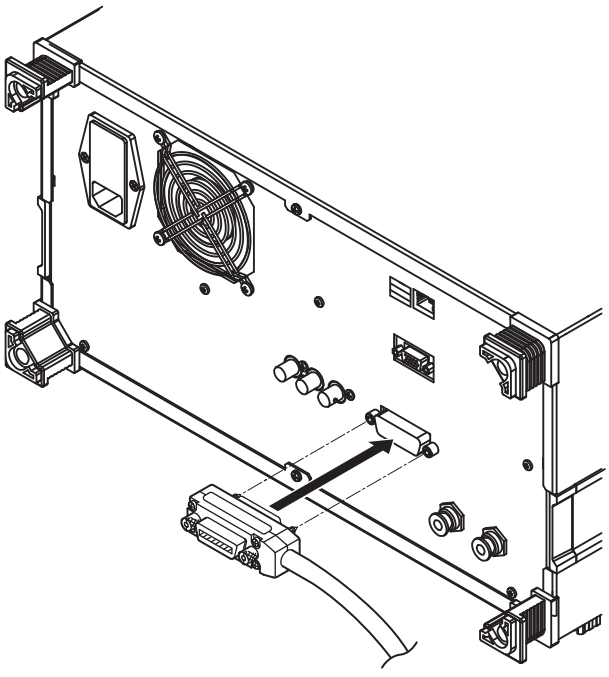
CAUTION

Always turn OFF the power to the instrument and the PC when connecting or disconnecting communication cables. Failure to turn OFF the power can result in malfunction or damage to internal circuitry.

French

ATTENTION

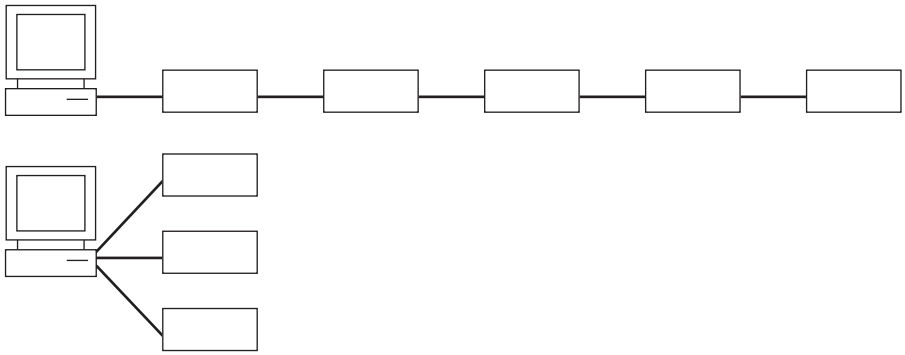
Veillez à mettre le PC et l'oscilloscope AQ6374E hors tension lorsque vous branchez ou débranchez les câbles de communication, car cela risquerait de provoquer des dysfonctionnements ou des courts-circuits internes.



AQ6374E

Precautions When Making Connections

- Securely fasten the screw that is attached to the GP-IB cable connector.
- You can connect several cables to connect to several devices. However, fifteen or more devices including the controller cannot be connected to a single bus.
- When connecting several devices, you cannot specify the same address for more than one.
- Use a cable of two meters or longer to connect between devices.
- Ensure that the total length in cables does not exceed twenty meters.
- When carrying out communications, make sure that at least two-thirds of all connected devices are turned ON.
- To connect multiple devices, wire them in a daisy-chain or star configuration as shown below. You can also mix these configurations. Loop configuration is not allowed.



2.2 GP-IB Interface Function

GP-IB Interface Function

Listener Function

- All of the same settings can be performed using the interface (except for power ON/OFF and communication settings) as when using the instrument's panel keys.
- Settings, waveforms, and other data can be received through output commands from the controller.
- Additionally, you can also receive commands regarding status reports and other data.

Talker Function

- Settings, waveforms, and other data can be output.

Note

Listen only, talk only, and controller functions are not available.

Switching between Remote and Local

Switching from Local to Remote

When in Local mode, if the instrument received a listen address from the controller that sets REN (remote enable) and ATN to "True," the instrument enters Remote mode.

- When in Remote mode, the REMOTE indicator lights.
- Keys other than the LOCAL key are disabled.
- Settings entered in Local mode are held even if switching to Remote mode.
- When an LLO (Local Lock Out) message is received from the controller, the instrument enters local lockout status. In LLO status, the LOCAL key is disabled and does not return this instrument to Local mode even when pressed. After cancelling the local lockout status, press the LOCAL key. To cancel the local lockout status, set REN to "False" from the controller.

Switching from Remote to Local

If you press the LOCAL key when in Remote mode the instrument enters Local mode. However, it does not return to Local mode if in the local lockout state.

- The REMOTE indicator turns off.
- All keys are enabled.
- Settings entered in Remote mode are held even if switching to Local mode.
- When a GTL (Go to Local) message is received from the controller, the instrument enters Local mode even if REN is set to False.

Note

The GP-IB interface cannot be used simultaneously with other communication interfaces (Ethernet).

2.3 GP-IB Interface Specifications

GP-IB Interface Specifications

Electromechanical specifications:	Conforms to IEEE std. 488-1978
Functional specifications:	See table below
Protocols:	Conforms to IEEE std. 488.2-1992
Encoding:	ISO (ASCII)
Mode:	Addressable mode
Address setting:	Addresses 0-30 can be set in the GP-IB setting screen in the SYSTEM menu.
Remote mode cancel:	Press LOCAL to cancel Remote mode. Note that this is disabled when under Local Lockout by the controller.

Functional Specifications

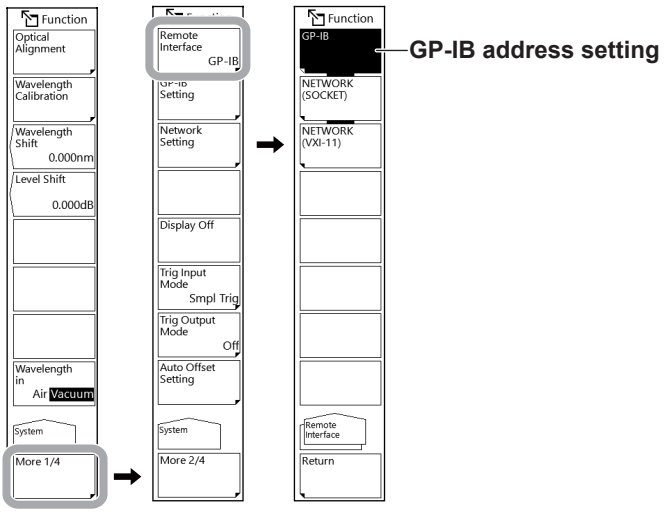
Function	Subset	Description
Source handshake	SH1	All capabilities of send handshake
Acceptor handshake	AH1	All capabilities of receive handshake
Talker	T6	Basic talker function, serial polling, and talker cancel function through MLA (my listen address). Talker only not provided.
Listener	L4	Basic listener function, serial polling, and listener cancel function through MLA (my listen address). Listener only not provided.
Service request	SR1	All service request functions
Remote local	RL1	All Remote/Local functions
Parallel port	PP0	Parallel polling function not provided
Device clear	DC1	All device clear functions
		Output buffer clear
		Input buffer clear (clearing of an unexecuted commands)
		Error buffer clear
		STB, ESR clear
Device trigger	DT0	Device trigger function
Controller	C0	Controller function not provided
Electrical characteristics	E1	Open collector

2.4 Setting the GP-IB Address

Procedure

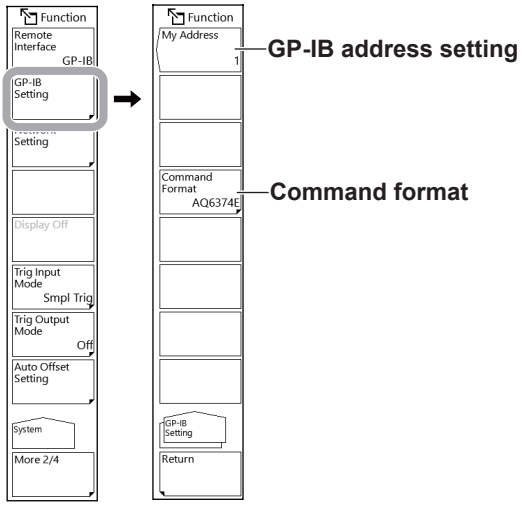
Selecting the Communication Interface

1. Press **SYSTEM**. The SYSTEM menu is displayed.
2. Tap the **More** repeatedly until the More 2/4 menu is displayed.
3. Tap the **Remote Interface**. The Remote Interface menu is displayed.
4. Tap the **GP-IB** to specify GP-IB as the communication interface.



Setting the Address

5. Tap the **GP-IB Setting**. The GP-IB Setting menu is displayed.
6. Tap the **My Address**. The GP-IB address setting screen is displayed.
7. Set the GP-IB address using the rotary knob or the arrow keys, and press ENTER.



Setting the Command Format

8. Perform these steps if you will use AQ6317 commands. Tap the **Command Format**. The command format setting menu is displayed.

9. Normally, you will enter AQ6374E. If you wish to use AQ6317 commands, enter AQ6317.

Explanation

The settings below are used when entering the settings that can be entered using the instrument's panel keys from a controller, or when outputting settings or waveform data to the controller.

GP-IB Address Settings

When in Addressable mode, set the instrument's address within the following range.

0–30

Each device that can be connected via GP-IB has its own unique GP-IB address. This address allows each device to be distinguished from other devices. Therefore, when connecting the instrument to a PC or other device, make sure not to set the same address on the instrument as any of the other devices.

Note

Do not change an address while the controller or other devices are using GP-IB.

Command Format Settings

Normally, you will enter AQ6374E mode.

If you wish to use the commands of the AQ6317 (another product in the series), enter AQ6317. See the appendix for AQ6317 commands that are compatible with the AQ6317.

2.5 Responses to Interface Messages

Responses to Interface Messages

Responses to Uniline Messages

- **IFC (Interface Clear)**
  Clears talker and listener. Output is cancelled if outputting data.
- **REN (Remote Enable)**
  Switches between Local and Remote.

  IDY (Identify) is not supported.

Responses to Multiline Messages (Address Commands)

- **GTL (Go To Local)**
  Switches to Local mode.
- **SDC (Selected Device Clear)**
  - Clears program messages (commands) being received, and the output queue.
  - The *OPC and *OPC? commands are invalid during execution.
  - The *WAI command closes immediately.

  PPC (parallel poll configure) and TCT (take control) are not supported.

Responses to Multiline Messages (Universal Commands)

- **LLO (Local Lockout)**
  Disables the front panel UNDO/LOCAL operation, and prohibits switching to Local mode.
- **DCL (Device Clear)**
  Same operation as SDC.
- **SPE (Serial Poll Enable)**
  Places the talker function of all devices on the bus in Serial poll mode. The controller polls each device in order.
- **SPD (Serial Poll Disable)**
  Cancels Serial poll mode for the talker function of all devices on the bus.

  PPU (Parallel Poll Unconfigure) is not supported.

Definition of Interface Messages

Interface messages are also called interface commands or bus commands, and are commands that are issued from the controller. Interface messages come in the following categories.

Uniline Messages

A message is sent through a single command line. The following are the three types of uniline messages.

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

Multiline Messages

A message is sent through eight data lines. Multiline messages come in the following categories.

- **Address Commands**

  These commands are valid when the device is specified as the listener or the talker. The following are the five types of address commands.

  Commands valid for devices specified as listeners

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

  Commands valid for devices specified as talkers

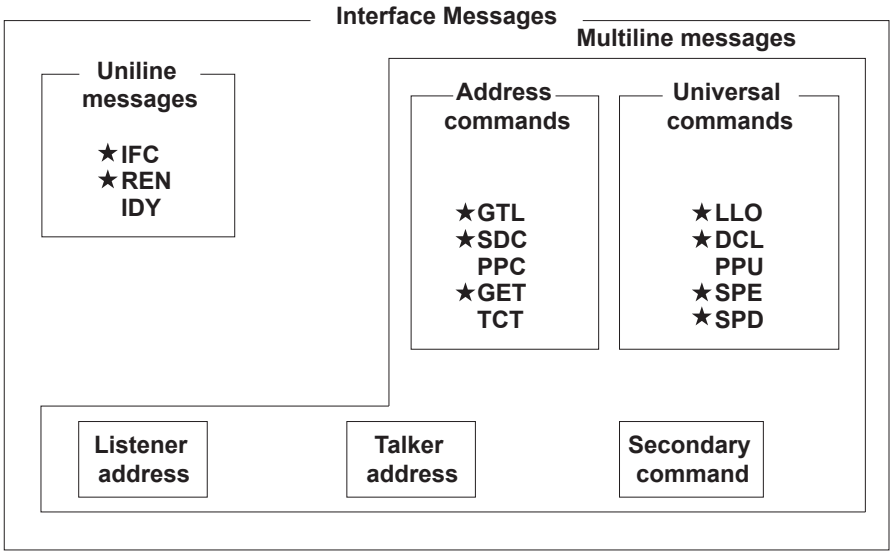
  - TCT (Take Control)

- **Universal Commands**

  These commands are valid for all devices regardless of whether they are specified as listeners, talkers, or neither. The following are the three types of universal commands.

  - LLO (Local Lockout)
  - DCL (Device Clear)
  - PPU (Parallel Poll Unconfigure)

Additionally, an interface message can consist of a listener address, talker address, or secondary command.



3.1 Connecting via Ethernet

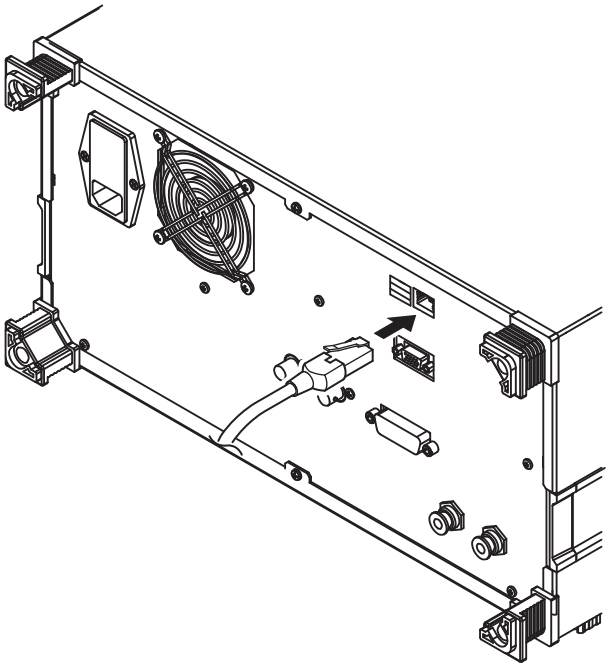
You can connect to a LAN using the Ethernet interface for control of the instrument from a PC.

Ethernet Interface Specifications

Communication ports:	1
Electromechanical specifications:	Conforms to IEEE802.3
Transmission method:	Ethernet (10BASE-T/100BASE-TX/1000BASE-T)
Transmission speed:	10 Mbps/100 Mbps/1000 Mbps
Communication protocol:	TCP/IP
Connector type:	RJ45
Port number used:	10001/tcp (default)

Connections

Connect a UTP (unshielded twisted-pair) cable or an STP (shielded twisted-pair) cable that is connected to another device to the ETHERNET port on the rear panel of the instrument.



AQ6374E

Precautions When Making Connections

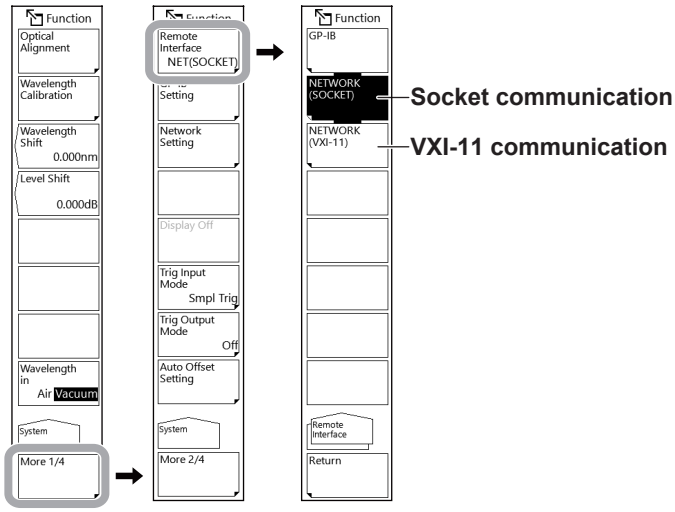
- Be sure to use a straight cable through a hub when connecting a PC to the instrument. Performance cannot be guaranteed if a 1-to-1 connection is made with a cross cable.
- Use a network cable that supports the data rate of your network.

3.2 Setting Up Ethernet

Procedure

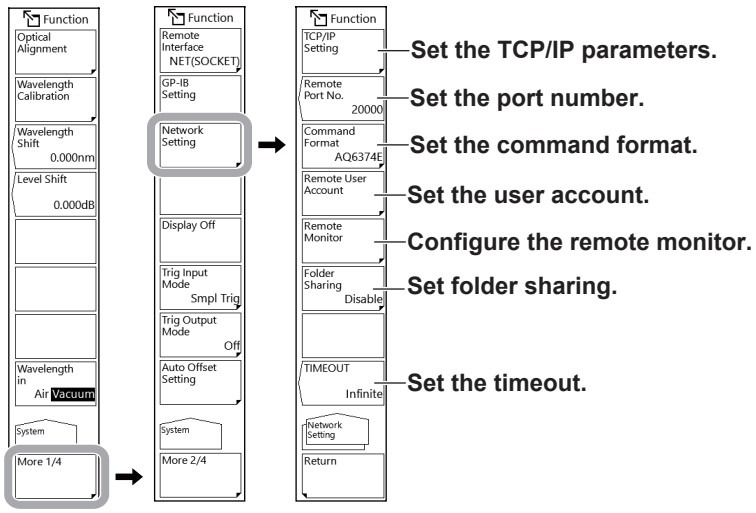
Configuring the communication interface

1. Press **SYSTEM**. The SYSTEM menu appears.
2. Tap **More** to display the More 2/4 menu.
3. Tap **Remote Interface**. The Remote Interface menu appears.
4. Tap **NETWORK (SOCKET)** or **NETWORK (VXI-11)** to set the communication interface to Ethernet.



Configuring the network settings

1. Press **SYSTEM**. The SYSTEM menu appears.
2. Tap **More** to display the More 2/4 menu.
3. Tap **Network Setting**. The Network Setting menu appears.



Setting the TCP/IP parameters

4. Tap **TCP/IP Setting**. The TCP/IP setting menu appears.

   Configure IPv4 and IPv6 according to the network you are using.

Setting the IPv4

5. Tap **AUTO (DHCP)** or **MANUAL**.

   To use a DHCP server, select AUTO (DHCP).

6. If you select MANUAL, set the IP address, subnet mask, and default gateway. If you select AUTO, proceed to step 8.

   Tap the IP address, subnet mask, and default gateway input boxes. A setting window appears.

7. Set the IP address, subnet mask, and default gateway on the setting window.

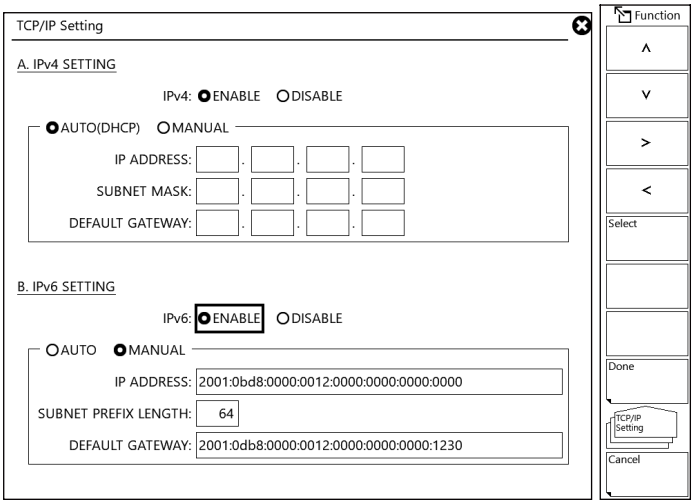
Setting the IPv6

5. Tap **AUTO** or **MANUAL**.
6. If you select MANUAL, set the IP address, subnet prefix length, and default gateway. If you select AUTO, proceed to step 8.

   Tap the IP address, subnet prefix length, and default gateway input boxes. A setting window appears.

7. Set the IP address, subnet prefix length, and default gateway on the setting window.

8. When you finish setting all the items, tap **Done**.

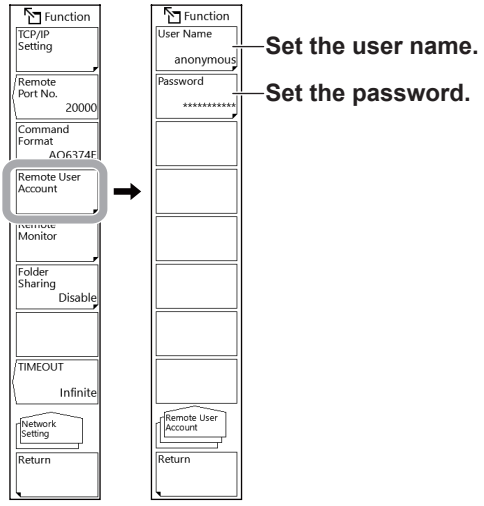


Setting the port number (not used with the VXI-11)

4. Continuing from step 3, tap **Remote Port No**. A window appears for setting the port number.
5. Set the port number on the setting window.

Setting the user account (not used with the VXI-11)

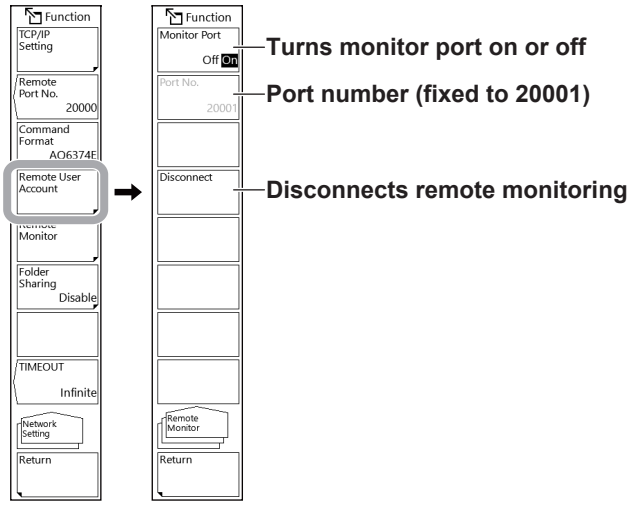
4. Continuing from step 3, tap **Remote User Account**. The Remote User Account menu appears.
5. Tap **User Name**. A keyboard appears.
   The default value is anonymous.
6. Enter the user name using up to 11 alphanumeric characters. If the user name is set to anonymous, you do not have to set a password.
7. Tap **Password**. A keyboard appears.
8. Enter the password using up to 11 alphanumeric characters.



Configuring the Remote Monitor

This function can be used to monitor the instrument's screen and control the instrument from a remote PC that is connected over TCP/IP.

4. Continuing from step 3, tap **Remote Monitor**. The remote monitor setting menu appears.
5. Tap **Monitor Port**. The value toggles between On and Off. A remote monitor can be connected when the setting is on.
6. To disconnect the remote monitor, tap **Disconnect**. The monitor will be disconnected from the PC.



Setting folder sharing

The user area folder of the instrument's internal memory can be shared with a PC.

4. Continuing from step 3, tap **Folder Sharing**. The Folder Sharing menu appears.
5. Tap **Read Only**. The user area folder of the instrument is shared over the network (read only).

   Tapping **Disable** will disable folder sharing.

   The user name and password needed to access the shared folder are as follows:

   User name: user

   Password: yokogawa

Setting the remote timeout (not used with the VXI-11)

4. Continuing from step 3, tap Time Out. A setting window appears.
5. Set the timeout value on the setting window.

Explanation

Configuring the network settings

Setting the TCP/IP parameters

Configure IPv4 and IPv6 according to the network that this instrument is connected to.
If both IPv4 and IPv6 are enabled, either one will be selected automatically according to the network that this instrument is connected to.

- **IPv4**
  If a DHCP server is available on the network that the instrument is connected to, an IP address is automatically assigned to the instrument. In this case, select AUTO (DHCP) in the TCP/IP settings.
  For details about the network that you intend to connect the instrument to, contact your network administrator.
- **IPv6**
  "AUTO" is usually used, but you can also set a fixed IP address manually.
  If you are setting the IP address manually, set also the subnet prefix length and default gateway.
  Set the IP address and default gateway in hexadecimal notation.

REMOTE PORT NO. (not used with the VXI-11)

Set the port number to use to control the instrument remotely over the network (socket). (Default value: 10001)

User authentication (not used with the VXI-11)

If you want to connect to the instrument from your PC over an Ethernet network, user authentication is required. If the user name is anonymous, you do not have to enter a password.
The instrument supports plaintext authentication and the MD5 algorithm (RSA Data Security, Inc. MD5 Message DigestAlgorithm).

Remote monitoring

This function can be used to monitor the instrument's screen and control the instrument from a remote PC that is connected to the instrument through a TCP/IP port.
To use this feature, you need remote monitoring software (not included). You cannot use the remote monitoring port to perform remote control with normal remote commands.
For information on remote monitoring software, contact your nearest YOKOGAWA dealer.

User name and password

A user name and password are required to access the instrument using this function.
Enter the user name using up to 11 alphanumeric characters. If the user name is set to anonymous, you do not have to set a password.
Enter the password using up to 11 alphanumeric characters.

MONITOR PORT

Enables and disables the remote monitor TCP/IP port.
If you set this to Off, the remote monitor will be disabled.

PORT NO.

This is the remote monitor TCP/IP port number. It is fixed to 20001. You cannot use this port to perform remote control with normal remote commands.

DISCONNECT

If you tap this menu while remote monitoring is connected, remote monitoring with the external PC will be disconnected. This menu can only be used when remote monitoring is connected.

Folder sharing

The user area folder of the instrument's internal memory can be shared with a PC. When the user area folder is shared, files in the folder can be copied to the PC over a network. Note that you cannot save files to the instrument.

If you are prompted to enter a user name and password when accessing the instrument's shared folder from the PC, enter the following information.

User name: user

Password: yokogawa

Timeout period (not used with the VXI-11)

If no communication takes place for the specified time in remote mode, the communication is automatically disconnected, and the mode changes to local.

When you change the timeout period, the elapsed time is reset.

You can set the time to Infinite (0 s) or between 1 and 21600 s (6 hours).

Remote Control Using Commands

The AQ6374E can be remote controlled using the LAN port.
For remote commands, use the same commands as those for control via the GP-IB interface.
The instrument also supports VXI-11 control.

Switching Interfaces

Select GP-IB, NETWORK (SOCKET), or NETWORK (VXI-11) as an interface to use for remote control. When you change the interface, the connection status is reset. In addition, the UNDO/LOCAL key disconnects the connection. Otherwise, the connection is kept open unless closed by the controller.

Remote Commands

As with GP-IB-based remote control, you can select the command format from the AQ6374E mode or from the AQ6317-compatible mode.

Interrupt by SRQ

An SRQ interrupt does not occur during NETWORK(SOCKET)-based remote control.

Status Register

The status registers operate in the same manner as in remote control via the GP-IB interface.
Using the “*SPOOL?” command dedicated for remote control using the LAN port allows you to read the status registers, as in the case with serial polling via the GP-IB interface.

- *STB?: When AQ6374E is the setting of the COMMAND FORMAT key
- SPOLL?: When AQ6317 is the setting of the COMMAND FORMAT key

Delimiter

The delimiter for LAN-based remote control is fixed to CR + LF.

Transmission of Talker Data

When NETWORK (SOCKET) is in use, if the instrument receives talker data from the controller PC, the instrument sends the data to the controller PC’s buffer. It receives the external PC's buffer data and stores the query data.

Connection

The instrument can only be connected to one controller (an external PC or other device). If the instrument receives a connection request from a controller while already connected to another controller, the new connection is not opened and the existing connection is kept open.

Computer Name

The instrument’s computer name is as follows.
For the AQ6374E
“6374E@@@@@@@@@” (where “@@@@@@@@@” is the serial number)

Commands that are Necessary for Remote Control over the LAN

The authentication by OPEN command is required to remote control over the NETWORK(SOCKET). Both the OPEN and CLOSE commands are also valid in AQ6317 mode.

OPEN

Function Sends the user name and starts user authentication.

Syntax `OPEN<wsp>"username"`

username = the user name

Example `OPEN "yokogawa" -> AUTHENTICATE CRAM-MD5.`

Explanation Authentication is carried out with the OPEN command as follows.

- **For Plain Text Authentication**

1. Send OPEN "username" to the AQ6374E. The response message is received from the AQ6374E.
2. Confirm that the received message is "AUTHENTICATE CRAM-MD5."
3. Send the password to the AQ6374E (anything can be input if the user name is anonymous).
4. If the message, "READY" is received from the AQ6374E, authentication was successful. The AQ6374E's REMOTE indicator lights, and sending of remote commands is enabled. If the user name and password are incorrect, authentication fails and the connections is closed.

- **For Encrypted Authentication**

1. Send OPEN "username" to the AQ6374E. The response message is received from the AQ6374E.
2. Confirm that the received message is "AUTHENTICATE CRAM-MD5."
3. Send "AUTHENTICATE CRAM-MD5 OK" to the AQ6374E. The response message (challenge string) is received from the AQ6374E.
4. The received challenge string and password are processed with an MD5 hash algorithm (anything can be input if the user name is anonymous).
5. Send the returned hash data (as a 32-character hexadecimal string in lower case) to the AQ6374E, and receive the response message.
6. If the message, "READY" is received from the AQ6374E, authentication was successful. The AQ6374E's REMOTE indicator lights, and sending of remote commands is enabled. If the user name and password are incorrect, authentication fails and the connection is closed.

CLOSE

Function Closes the connection (turns it OFF), and switches to local mode.

Syntax `CLOSE`

Example `CLOSE`

4.1 Rules of Syntax and Command Types

The following information is intended for the common commands and instrument-specific commands contained in this manual. Measured values and parameters are all sent and received using ASCII characters, not including special commands.

Description of Rules of Syntax

Rule	Description
|	Indicates that one of the elements in a list should be selected.
	E.g.: `A|B|C` = A, B, or C is used
[ ]	An item in square brackets is specified as desired.
<wsp>[1]	Space
<integer>	Integer
<NRf>	Exponent indicating value
<"file name">	A file name can be a maximum of 56 characters, including extensions, excluding the directory part. Enclose a character string using double quotations (" ").
<trace name>	Trace name (TRA|TRB|TRC|TRD|TRE|TRF|TRG)
<marker>	Marker number (0: moving marker, 1 to 1024: fixed markers)
<"string">	Character string Enclose a character string using double quotations (" ").

1. Regarding white space (<wsp>):
   White space is defined as a character corresponding to 00h to 20h (not including 0Ah (LF)) of the ASCII character sets. Aside from inserting it between a command and parameters (when specifying parameters) or using it as space in a character string such as a file name in a parameter, white space can be inserted as desired to make a program legible.

Collective Transmission of Multiple Commands

You can create a command string using the commands described in chapter 5, “Commands” and send it to the instrument. If multiple commands are written in a single output statement by using a semicolon “;” to delimit each command, the commands will be executed in the order in which they have been written.

Format of a Remote Command

- **Short and Long Forms**
  The instrument’s commands support both short and long forms.
  For the commands contained in this manual, the part written in capital letters is the short form of the command concerned. The short form of the INITiate command is INIT.

- **Upper- and Lower-Case Letters**
  The instrument does not distinguish between upper- and lower-case letters.
  Return values are all in upper-case letters.

- **Grouping of SCPI Commands Using a Subsystem**

  The instrument supports the subsystem-based grouping of the SCPI commands.
  Commands belonging to the same sub-system and existing at the same tree of the hierarchical structure of the subsystem can be sent in combination. In this case, each command should be delimited by a semicolon.
  List of GP-IB commands used in examples

```
:SENSe   :SETTing
                :SMOothing
         :WAVelength
                :STOP
                :STARt
:SENSe:WAVelength:STARt 1500NM;STOP 1600NM    (Y)
:SENSe:WAVelength:STARt 1500NM;SMOothing ON    (X)
         (Reason: They are not in the same hierarchy.)
:SENSe:WAVelength:STARt 1500NM;:STOP 1600NM      (X)
         (Reason: A colon ":" is unnecessary after a semicolon ";".)
```

- **Numerics**

  This instrument supports multiple notation methods when receiving a numeric(s).
  This instrument uses only the basic units when transmitting a numeric(s).
  The number of digits for the real part is fixed to a one digit integer (with a sign) and eight digits for decimal places. The number of digits for the exponential part is fixed to 3.

  Ex.: Receivable numerics (in case of 1550 nm)

  1550 nm, 1.55 um, 1550E-9, 1.55E-6, and others

  Ex.: Transmittable numerics (in case of 1550 nm)

  +1.55000000E-006 only

  If a received numeric has a precision higher than the range of numerics handled inside this unit, lower decimal places will be rounded off rather than being discarded.

  This instrument can handle the following multiplier suffixes:

Multiplier	Mnemonic	Multiplier	Mnemonic
1E18	EX (exa)	1E-3	M (milli)
1E15	PE (peta)	1E-6	U (micro)
1E12	T (tera)	1E-9	N (nano)
1E9	G (giga)	1E-12	P (pico)
1E6	MA (mega)	1E-15	F (femto)
1E3	K (kilo)	1E-18	A (atto)

- **Specification of Parameters in a Command**

  To use parameters in a command, a space must be placed between the command and parameters. Each parameter is delimited by a comma ",". A space may also be placed before and after a comma to make the command legible.

4.2 Types of Commands

This unit's commands can be classified into the following three types:

Types of Commands

Sequential Commands

- These commands are the most general commands.
- The action of another command is not performed until the running of a sequential command is complete.
- Another action is not started until the running of the other command is complete.

Overlappable Commands

- An overlappable command allows execution of an overlapping command while it is being run.
  Ex. of command: :INITialte Makes a sweep.

Overlapping Commands

- An overlapping command can be executed while an overlappable command is being run.
- These commands cannot be executed while a sequential command is being executed or if it has not yet been processed.
  Ex. of command: :ABORt Stops measurement or calibration action.
  *STB? Reads status byte.

AQ6317-Compatible Commands

The instrument supports AQ6317-compatible commands. When using AQ6317-compatible commands, call up the SYSTEM menu using the SYSTEM key and place the instrument in AQ6317-compatible mode.

5.1 List of Commands

Command	Function	Page
ABORt Group		
:ABORt	Stops operations such as measurements and calibration.	5-9
APPLication Group		
:APPLication:DLOGging:ETIMe?	Queries the elapsed time of data logging (in seconds).	5-10
:APPLication:DLOGging:LPARameter:INTerval	Sets or queries the measurement interval of data logging.	5-10
:APPLication:DLOGging:LPARameter:ITEM	Sets or queries the data logging source.	5-10
:APPLication:DLOGging:LPARameter:LMODe	Sets or queries the data logging mode (maximum channel mode or maximum logging mode).	5-10
:APPLication:DLOGging:LPARameter:MEMory	Sets or queries the temporary area for saving waveform files of data logging.	5-10
:APPLication:DLOGging:LPARameter:MTHResh	Sets or queries the threshold of the channel-matching wavelength λ for data logging.	5-11
:APPLication:DLOGging:LPARameter:PDETect:ATHResh	Sets or queries the threshold (absolute value) for detecting the data logging mode.	5-11
:APPLication:DLOGging:LPARameter:PDETect:RTHResh	Sets or queries the threshold (relative value) for detecting the data logging mode.	5-11
:APPLication:DLOGging:LPARameter:PDETect:TTYPe	Sets or queries how the threshold for detecting the data logging mode (peak or bottom) is specified.	5-11
:APPLication:DLOGging:LPARameter:TDURation	Sets or queries the measurement duration of data logging (in seconds).	5-11
:APPLication:DLOGging:LPARameter:TLOGging	Sets or queries whether waveforms will be logged during data logging.	5-11
:APPLication:DLOGging:STATe	Starts, stops, or queries data logging.	5-12
:APPLication:PROGram:EXECute	Executes a program that has been specified.	5-12
CALCulate Group		
:CALCulate:ARESolution?	Queries the actual resolution data of the specified trace.	5-13
:CALCulate:CATegory	Sets or queries the type of analysis.	5-14
:CALCulate:DATA?	Queries the analysis results.	5-14
:CALCulate:DATA:CGAin?	Queries the gain value of the EDFA-NF analysis results.	5-14
:CALCulate:DATA:CNF?	Queries the NF value of the EDFA-NF analysis results.	5-14
:CALCulate:DATA:COLor?	Queries the dominant wavelength of the measured light source and the x coordinate, y coordinate, z coordinate, color temperature, and deviation of the measured light source on the xy chromaticity diagram.	5-15
:CALCulate:DATA:CPOWers?	Queries the level value of the OSNR(WDM), WDM, EDFA-NF, WDM FIL-PK, or WDM FIL-BTM analysis results.	5-15
:CALCulate:DATA:CSNR?	Queries the SNR value from the last time WDM analysis was executed.	5-15
:CALCulate:DATA:CWAVelengths?	Queries the wavelength value of the OSNR(WDM), WDM, EDFA-NF, WDM FIL-PK, or WDM FIL-BTM analysis results.	5-15
:CALCulate:DATA:DFBLd?	Queries the DFB-LD analysis results.	5-16
:CALCulate:DATA:NCHannels?	Queries the number of channels of the OSNR(WDM), WDM, EDFA-NF, WDM FIL-PK, WDM FIL-BTM, or WDM SMSR analysis results.	5-16
:CALCulate:DATA:OSLope?	Queries the OUTPUT SLOPE value of the OSNR (WDM) and WDM analysis results.	5-16
:CALCulate:DISPlay	Sets or queries the display format of analysis results.	5-16
:CALCulate:DISPlay:GRAPh:LMARKer:Y	Sets or queries the position of line marker Y1 or Y2 on the graph display of analysis results.	5-16
:CALCulate[:IMMediate]	Executes analysis. Queries the result of whether analysis has been performed.	5-16
:CALCulate[:IMMediate]:AUTO	Sets or queries the automatic analysis function.	5-17
:CALCulate:LMARker:AOFF	Clears all line markers.	5-17
:CALCulate:LMARker:SRANge	Sets or queries whether to limit an analytical range to the spacing between line markers L1 and L2.	5-17
:CALCulate:LMARker:SSPan	Sets spacing between line markers L1 and L2 for span.	5-17

5.1 List of Commands

Command	Function	Page
:CALCulate:LMARker:SZSPan	Sets spacing between line markers L1 and L2 for zoom span.	5-17
:CALCulate:LMARker:X	Sets or queries the position of line markers L1 and L2.	5-17
:CALCulate:LMARker:Y	Sets or queries the position of line markers L3 and L4.	5-17
:CALCulate:MARKer:AOFF	Clears all markers.	5-17
:CALCulate:MARKer:AUTO	Sets or queries the auto search function.	5-17
:CALCulate:MARKer:FUNCtion:FORMat	Sets the format of a difference value displayed in the area marker and queries the format set.	5-18
:CALCulate:MARKer:FUNCtion:UPDate	Sets or queries ON/OFF of the automatic update function of fixed markers used when updating an active trace.	5-18
:CALCulate:MARKer:IRANge	Sets or queries the integration frequency range of the specified integral marker.	5-18
:CALCulate:MARKer:MAXimum	Detects a peak and places the moving marker on that peak.	5-18
:CALCulate:MARKer:MAXimum:LEFT	Detects the nearest peak existing on the left side of the current position of the moving marker and places the moving marker on that peak.	5-18
:CALCulate:MARKer:MAXimum:NEXT	Detects the highest peak that is below the level of the current position of the moving marker and places the moving marker on that peak.	5-18
:CALCulate:MARKer:MAXimum:RIGHt	Detects the nearest peak existing on the right side of the current position of the moving marker and places the moving marker on that peak.	5-19
:CALCulate:MARKer:MAXimum:SCENter	Detects the peak wavelength and sets it as the measurement center waveform.	5-19
:CALCulate:MARKer:MAXimum:SCENter:AUTO	Sets or queries ON/OFF of the function to automatically detect the peak wavelength and set it as the measurement center wavelength.	5-19
:CALCulate:MARKer:MAXimum:SRLevel	Detects the peak level and sets it for the reference level.	5-19
:CALCulate:MARKer:MAXimum:SRLevel:AUTO	Sets or queries ON/OFF of the function to automatically detect the peak level and sets it as the reference level.	5-19
:CALCulate:MARKer:MAXimum:SZCenter	Detects the peak wavelength and sets it as the display center wavelength.	5-19
:CALCulate:MARKer:MINimum	Detects the bottom and places the moving marker on that bottom.	5-19
:CALCulate:MARKer:MINimum:LEFT	Detects the nearest bottom existing on the left side of the current position of the moving marker and places the moving marker on that bottom.	5-19
:CALCulate:MARKer:MINimum:NEXT	Detects the lowest bottom that is above the level of the current position of the moving marker and places the moving marker on that bottom.	5-19
:CALCulate:MARKer:MINimum:RIGHt	Detects the nearest bottom existing on the right side of the current position of the moving marker and places the moving marker on that side.	5-20
:CALCulate:MARKer:MSEarch	Sets or queries the type of the search function.	5-20
:CALCulate:MARKer:MSEarch:SORT	Sets or queries the sort order of the multi search detection list.	5-20
:CALCulate:MARKer:MSEarch:THResh	Sets or queries the multi search threshold.	5-20
:CALCulate:MARKer{:PDENsity|:NOISe}{:BWIDth|:BANDwidth}	Sets or queries the normalization bandwidth of the specified power spectral density marker.	5-20
:CALCulate:MARKer:SCENter	Sets the wavelength of the current moving marker as the measurement center waveform.	5-20
:CALCulate:MARKer:SRLevel	Sets the current level of the moving marker for the reference level.	5-20
:CALCulate:MARKer[:STATe]	Specified marker is positioned or deleted in the position of the moving marker. Also, queries the status of the specified marker.	5-21
:CALCulate:MARKer:SZCenter	Sets the current wavelength of the moving marker for the display center wavelength.	5-21
:CALCulate:MARKer:TYPE	Sets or queries the marker type.	5-21
:CALCulate:MARKer:UNIT	Sets or queries the units of display for the marker values.	5-21
:CALCulate:MARKer:X	Sets the specified marker at the specified position.Queries the X value of the specified marker.	5-21
:CALCulate:MARKer:Y?	Queries the Y value of the specified marker.	5-22
:CALCulate:MATH:TRC	Sets or queries the TRACE C calculation function.	5-22
:CALCulate:MATH:TRC:K	Sets or queries parameter K of the TRACE C calculation function.	5-22
:CALCulate:MATH:TRF	Sets or queries the TRACE F calculation function.	5-22
:CALCulate:MATH:TRF:PNBW:BWIDth|BANDwidth	Sets or queries the normalization bandwidth of the power spectral density trace.	5-22
:CALCulate:MATH:TRG	Sets or queries the TRACE G calculation function.	5-22
:CALCulate:MATH:TRG:CVFT:FALGo	Sets or queries the fitting curve function of the TRACE G fitting curve function.	5-23
:CALCulate:MATH:TRG:CVFT:OPARea	Sets or queries a calculation area during curve fit and peak curve fit.	5-23
:CALCulate:MATH:TRG:CVFT:THResh	Sets or queries the threshold value for curve fitting.	5-23

Command	Function	Page
:CALCulate:MATH:TRG:PCVFt:THResh	Sets or queries the threshold value for peak curve fitting.	5-23
:CALCulate:PARameter[:CATegory]:DFBLd	Sets or queries parameters for the DFB-LD analysis function.	5-24
:CALCulate:PARameter[:CATegory]:FILBtm	Sets or queries parameters for the FILTER-BTM analysis function.	5-24
:CALCulate:PARameter[:CATegory]:FILPk	Sets or queries parameters for the FILTER PEAK analysis function.	5-25
:CALCulate:PARameter[:CATegory]:FPLD	Sets or queries parameters for the FP-LD analysis function.	5-25
:CALCulate:PARameter[:CATegory]:ITLa	Sets or queries parameters for the TLS analysis function.	5-26
:CALCulate:PARameter[:CATegory]:LED	Sets or queries parameters for the LED analysis function.	5-26
:CALCulate:PARameter[:CATegory]:NF:AALGo	Sets or queries the measurement algorithm applied to ASE level measurements made by the NF analysis function.	5-27
:CALCulate:PARameter[:CATegory]:NF:FALGo	Sets or queries the fitting function during level measurement applied to ASE level measurements made by the NF analysis function.	5-27
:CALCulate:PARameter[:CATegory]:NF:FARea	Sets or queries the fitting range for level measurement applied to ASE level measurements made by the NF analysis function.	5-27
:CALCulate:PARameter[:CATegory]:NF:IOFFset	Sets or queries level offset values (signal light) for the NF analysis function.	5-27
:CALCulate:PARameter[:CATegory]:NF:IRANge	Sets or queries the integration frequency range for when the EDFA-NF analysis feature calculates the signal optical power.	5-27
:CALCulate:PARameter[:CATegory]:NF:MARea	Sets or queries the mask range for level measurement applied to ASE level measurements made by the NF analysis function.	5-28
:CALCulate:PARameter[:CATegory]:NF:MDIFf	Sets or queries the peak bottom difference of channel detection for the NF analysis function.	5-28
:CALCulate:PARameter[:CATegory]:NF:OOFFset	Sets or queries level offset values (output light) for the NF analysis function.	5-28
:CALCulate:PARameter[:CATegory]:NF:PDISplay	Sets or queries whether to display data used for fitting of the NF analysis function on the waveform screen.	5-28
:CALCulate:PARameter[:CATegory]:NF:TH	Sets or queries the threshold level of channel detection for the NF analysis function.	5-28
:CALCulate:PARameter[:CATegory]:NF:RBWidth	Sets or queries the method for calculating the resolution value of the NF computation.	5-28
:CALCulate:PARameter[:CATegory]:NF:SNOise	Sets or queries whether Shot Noise is included in the NF computation	5-29
:CALCulate:PARameter[:CATegory]:NF:SPOWer	Sets or queries the signal optical power calculation method of the EDFA-NF analysis feature.	5-29
:CALCulate:PARameter[:CATegory]:NOTCh:K	Sets or queries the magnification of the notch width analysis function.	5-29
:CALCulate:PARameter[:CATegory]:NOTCh:TH	Sets or queries the threshold value for the notch width analysis function.	5-29
:CALCulate:PARameter[:CATegory]:NOTCh:TYPE	Sets or queries the analysis direction of the notch width analysis function.	5-29
:CALCulate:PARameter[:CATegory]:POWer:OFFSet	Sets or queries the offset value for the POWER analysis function.	5-29
:CALCulate:PARameter[:CATegory]:SMSR:BANDwidth|:BWIDth	Sets or queries the resolution bandwidth used for side mode power normalization of the SMSR analysis function.	5-30
:CALCulate:PARameter[:CATegory]:SMSR	Sets or queries the mask value for the SMSR analysis function.	5-30
:CALCulate:PARameter[:CATegory]:SMSR:MODE	Sets or queries the analysis mode for the SMSR analysis function.	5-30
:CALCulate:PARameter[:CATegory]:SMSR:MDIFf	Sets or queries the value for peak/bottom difference for the SMSR analysis function.	5-30
:CALCulate:PARameter[:CATegory]:SMSR:SMPower	Sets or queries the side mode power calculation method of the SMSR analysis function.	5-30
:CALCulate:PARameter[:CATegory]:SWENvelope:K	Sets or queries the magnification of the ENVELOPE method–based spectrum width analysis function.	5-30
:CALCulate:PARameter[:CATegory]:SWENvelope:TH1	Sets or queries the search threshold level of the ENVELOPE method–based spectrum width analysis function.	5-30
:CALCulate:PARameter[:CATegory]:SWENvelope:TH2	Sets or queries the threshold level of the ENVELOPE method–based spectrum width analysis function.	5-31

Command	Function	Page
:CALCulate:PARameter[:CATegory] :SWPKrms:K	Sets or queries the magnification of the PEAK-RMS method–based spectrum width analysis function.	5-31
:CALCulate:PARameter[:CATegory] :SWPKrms:TH	Sets or queries the threshold level of the PEAK-RMS method–based spectrum width analysis function.	5-31
:CALCulate:PARameter[:CATegory] :SWRMs:K	Sets or queries the magnification of the RMS method–based spectrum width analysis function.	5-31
:CALCulate:PARameter[:CATegory] :SWRMs:TH	Sets or queries the threshold level of the RMS method–based spectrum width analysis function.	5-31
:CALCulate:PARameter[:CATegory] :SWTHresh:K	Sets or queries the magnification of the THRESH method–based spectrum width analysis function.	5-31
:CALCulate:PARameter[:CATegory] :SWTHresh:MFIT	Sets or queries whether to enable the mode fit of the THRESH method-based spectrum width analysis function.	5-32
:CALCulate:PARameter[:CATegory] :SWTHresh:TH	Sets or queries the threshold level of the THRESH method–based spectrum width analysis function.	5-32
:CALCulate:PARameter[:CATegory] :WDM:DMASk	Sets or queries the channel mask threshold level for the WDM analysis function.	5-32
:CALCulate:PARameter[:CATegory] :WDM:DTYPe	Sets or queries the displayed waveforms of the analysis results for the WDM analysis function.	5-32
:CALCulate:PARameter[:CATegory] :WDM:DUAL	Sets or queries the SNR calculation mode for the WDM analysis function.	5-32
:CALCulate:PARameter[:CATegory] :WDM:FALGo	Sets or queries the fitting function during level measurement applied to noise level measurements made by the WDM analysis function.	5-33
:CALCulate:PARameter[:CATegory] :WDM:IRANge	Sets or queries the integral frequency range during signal light power calculation by the WDM analysis function	5-33
:CALCulate:PARameter[:CATegory] :WDM:IRANge:TH	Sets or queries the signal optical power calculation threshold of the WDM analysis function.	5-33
:CALCulate:PARameter[:CATegory] :WDM:MAREa	Sets or queries the mask range during level measurement applied to noise level measurements made by the WDM analysis function.	5-33
:CALCulate:PARameter[:CATegory] :WDM:MDIFf	Sets or queries the peak bottom difference of channel detection for the WDM analysis function.	5-33
:CALCulate:PARameter[:CATegory] :WDM:MMReset	Resets the maximum and minimum of the drift values of the WDM analysis function.	5-33
:CALCulate:PARameter[:CATegory] :WDM:NALGo	Sets or queries the measurement algorithm applied to noise level measurements made by the WDM analysis function.	5-34
:CALCulate:PARameter[:CATegory] :WDM:NAREa	Sets or queries the measuring range applied to noise level measurements made by the WDM analysis function.	5-34
:CALCulate:PARameter[:CATegory] :WDM:NBW	Sets or queries the noise bandwidth for the WDM analysis function.	5-34
:CALCulate:PARameter[:CATegory] :WDM:ODISplay	Sets or queries whether the OSNR analysis results of the WDM analysis function are shown.	5-34
:CALCulate:PARameter[:CATegory] :WDM:OSLope	Sets or queries whether to enable the function of obtaining the least square approximation line in the WDM analysis function.	5-34
:CALCulate:PARameter[:CATegory] :WDM:PDISplay	Sets or queries whether to display data used for fitting of the WDM analysis function on the waveform screen.	5-34
:CALCulate:PARameter[:CATegory] :WDM:RCH	Sets or queries the reference channel used in calculating the offset wavelength/level of the WDM analysis function.	5-35
:CALCulate:PARameter[:CATegory] :WDM:RELation	Sets or queries the display format of the wavelength/level relative values for the WDM analysis function.	5-35
:CALCulate:PARameter[:CATegory] :WDM:SPOWer	Sets or queries the signal light power calculation method of the WDM analysis function.	5-35
:CALCulate:PARameter[:CATegory] :WDM:TH	Sets or queries the threshold level of channel detection for the WDM analysis function.	5-35
:CALCulate:PARameter[:CATegory] :WDMSmsr:DMASk	Sets or queries the channel mask threshold of the WDM SMSR analysis function.	5-35
:CALCulate:PARameter[:CATegory] :WDMSmsr:MASK	Sets or queries the threshold of the WDM SMSR analysis function.	5-36
:CALCulate:PARameter[:CATegory] :WDMSmsr:MDIFf	Sets or queries the channel detection peak/bottom difference of the WDM SMSR analysis function.	5-36
:CALCulate:PARameter[:CATegory] :WDMSmsr:MODE	Sets or queries the SMSR analysis mode of the WDM SMSR analysis function.	5-36
:CALCulate:PARameter[:CATegory] :WDMSmsr:SAREa	Sets or queries each channel's SMSR analysis range of the WDM SMSR analysis function.	5-36
:CALCulate:PARameter[:CATegory] :WDMSmsr:TH:	Sets or queries the channel detection threshold of the WDM SMSR analysis function.	5-36

Command	Function	Page
:CALCulate:PARameter[:CATegory]:WFBottom	Sets or queries parameters for the WDM FILTER-BTM analysis function.	5-37
:CALCulate:PARameter[:CATegory]:WFPeak	Sets or queries parameters for the WDM FILTER-PEAK analysis function.	5-37
:CALCulate:PARameter:COMMon:MDIFf	Sets or queries the peak-bottom difference parameter of channel detection used in the analysis function.	5-37

CALibration Group

:CALibration:ALIGn[:IMMediate]	Executes optical axis adjustment of the monochromator using the internal reference light source.	5-38
:CALibration:ALIGn:INTernal[:IMMediate]	Executes optical axis adjustment of the monochromator using the internal reference light source.	5-38
:CALibration:WAVelength:EXTernal[:IMMediate]	Performs wavelength calibration using an external reference light source.	5-38
:CALibration:WAVelength:EXTernal:SOURce	Sets or queries the type of the light source used for external reference light source-based wavelength calibration.	5-38
:CALibration:WAVelength:EXTernal:WAVelength	Sets or queries the wavelength of the light source used for external reference light source-based wavelength calibration.	5-38
:CALibration:WAVelength:INTernal[:IMMediate]	Performs wavelength calibration using an internal reference light source.	5-39
:CALibration:ZERO[:AUTO]	Sets or queries whether to enable the auto offset function of the level.	5-39
:CALibration:ZERO[:AUTO]:INTerval	Sets or queries the time interval for executing the Auto Offset function for the level.	5-39
:CALibration:ZERO[:AUTO]:STATus?	Queries the offset adjustment status.	5-39

DISPlay Group

:DISPlay:COLor	Sets or queries the screen color mode.	5-40
:DISPlay[:WINDow]	Sets or queries whether the display is enabled.	5-40
:DISPlay[:WINDow]:OVIew:POSition	Sets or queries the ON/OFF and position of the OVERVIEW display shown during zoom operation.	5-40
:DISPlay[:WINDow]:TEXT:CLEar	Clears labels.	5-40
:DISPlay[:WINDow]:TEXT:DATA	Sets or queries the labels.	5-40
:DISPlay[:WINDow]:TRACe:X[:SCALe]:CENTer	Sets or queries the center wavelength of the X-axis of the display scale.	5-40
:DISPlay[:WINDow]:TRACe:X[:SCALe]:INITialize	Initializes the X-axis parameters of the display scale.	5-40
:DISPlay[:WINDow]:TRACe:X[:SCALe]:SMSCale	Sets parameters of the current display scale to the measurement scale.	5-41
:DISPlay[:WINDow]:TRACe:X[:SCALe]:SPAN	Sets or queries the span of the X-axis of the display scale.	5-41
:DISPlay[:WINDow]:TRACe:X[:SCALe]:SRANge	Sets or queries whether to limit an analytical range to the display scale range.	5-41
:DISPlay[:WINDow]:TRACe:X[:SCALe]:STARt	Sets or queries the start wavelength of the X-axis of the display scale.	5-41
:DISPlay[:WINDow]:TRACe:X[:SCALe]:STOP	Sets or queries the stop wavelength of the X-axis of the display scale.	5-41
:DISPlay[:WINDow]:TRACe:Y:NMASk	Sets whether to mask the display of waveforms the level of which is at or below a set threshold level or queries the condition of whether the relevant waveform display is masked.	5-41
:DISPlay[:WINDow]:TRACe:Y:NMASk:TYPE	Sets or queries the display method when a waveform display at or below a threshold level is masked.	5-42
:DISPlay[:WINDow]:TRACe:Y[:SCALe]:DNUMber	Sets or queries the number of display divisions of the level axis.	5-42
:DISPlay[:WINDow]:TRACe:Y1[:SCALe]:BLEVel	Sets or queries the base level applied when the main scale of the level axis is linear.	5-42
:DISPlay[:WINDow]:TRACe:Y1[:SCALe]:PDIVision	Sets or queries the main scale of the level axis.	5-42
:DISPlay[:WINDow]:TRACe:Y1[:SCALe]:RLEVel	Sets or queries the reference level of the main scale of the level axis.	5-42
:DISPlay[:WINDow]:TRACe:Y1[:SCALe]:RPOSition	Sets or queries the position of the reference level of the main scale of the level axis.	5-43

Command	Function	Page
:DISPlay[:WINDow]:TRACe:Y1[:SCALe]:SPACing	Sets or queries the scale mode of the main scale of the level axis.	5-43
:DISPlay[:WINDow]:TRACe:Y1[:SCALe]:UNIT	Sets or queries the units of the main scale of the level axis.	5-43
:DISPlay[:WINDow]:TRACe:Y2[:SCALe]:AUTO	Sets or queries the automatic setting function of the sub scale of the level axis.	5-43
:DISPlay[:WINDow]:TRACe:Y2[:SCALe]:LENGth	Sets or queries the parameter of the optical fiber length used when the unit of the subscale of the level axis is dB/km.	5-43
:DISPlay[:WINDow]:TRACe:Y2[:SCALe]:OLEVel	Sets or queries the offset level of the sub scale of the level axis.	5-43
:DISPlay[:WINDow]:TRACe:Y2[:SCALe]:PDIVision	Sets or queries the sub scale of the level axis.	5-44
:DISPlay[:WINDow]:TRACe:Y2[:SCALe]:RPOSition	Sets or queries the position of the reference level of the sub scale of the level axis.	5-44
:DISPlay[:WINDow]:TRACe:Y2[:SCALe]:SMINimum	Sets or queries the value of the bottom of the scale applied when the subscale of the level axis is set to the linear or % mode.	5-44
:DISPlay[:WINDow]:TRACe:Y2[:SCALe]:UNIT	Sets or queries the units of the sub scale of the level axis.	5-44
:DISPlay[:WINDow]:TRACe:Y1[:SCALe]:INITialize	Initializes the display zoom condition of the main level scale.	5-44
:DISPlay[:WINDow]:TRACe:Y2[:SCALe]:INITialize	Initializes the display zoom condition of the sub level scale.	5-44

FORMat Group

:FORMat[:DATA]	Sets or queries the format used for data transfer.	5-45

INITiate Group

:INITiate[:IMMediate]	Makes a sweep.	5-46
:INITiate:SMODe	Sets or queries the sweep mode.	5-46

MMEMory Group

:MMEMory:ANAMe	Sets or queries the naming rule for automatic file names.	5-47
:MMEMory:CATalog?	Queries a list of all files in the current directory.	5-47
:MMEMory:CDRive	Sets or queries the current drive.	5-47
:MMEMory:COPY	Copies a specified file.	5-47
:MMEMory:DATA?	Queries the data in the specified file.	5-47
:MMEMory:DELete	Deletes a specified file.	5-48
:MMEMory:LOAD:ATRace	Loads the specified waveform files (all traces) into traces.	5-48
:MMEMory:LOAD:DLOGging	Loads the specified data logging file.	5-48
:MMEMory:LOAD:PROGram	Loads a specified program file into a specified program number.	5-48
:MMEMory:LOAD:SETTing	Loads a specified setting file.	5-48
:MMEMory:LOAD:TRACe	Loads a specified waveform file into a specified trace.	5-48
:MMEMory:MDIRectory	Creates a new directory.	5-48
:MMEMory:REMove	Readies the USB storage media for removal or queries the readiness status.	5-48
:MMEMory:REName	Renames a specified file.	5-48
:MMEMory:STORe:ARESult	Stores a variety of analysis results to a specified file.	5-48
:MMEMory:STORe:ATRace	Stores the specified waveform files (all traces) into traces.	5-49
:MMEMory:STORe:ATRace:TYPE	Sets or queries the file type for saving all trace waveform data.	5-49
:MMEMory:STORe:DATA	Stores a variety of data to a specified file.	5-49
:MMEMory:STORe:DATA:ITEM	Sets or queries an item to be used when storing data.	5-49
:MMEMory:STORe:DATA:MODE	Sets whether to insert data into or overwrite an existing file with the data when storing it or queries the condition of whether data is inserted or overwritten.	5-49
:MMEMory:STORe:DLOGging	Saves the data logging results to a specified file.	5-49
:MMEMory:STORe:DLOGging:CSAVe	Sets or queries whether data logging results will be saved to a file in CSV format.	5-49
:MMEMory:STORe:DLOGging:TSAVe	Sets or queries whether temporary saved waveform files will be saved when data logging results is saved.	5-49
:MMEMory:STORe:GRAPhics	Stores a waveform screen to a specified graphic file.	5-50
:MMEMory:STORe:PROGram	Stores a specified program to a specified file.	5-50
:MMEMory:STORe:SETTing	Stores setting information to a specified file.	5-50
:MMEMory:STORe:TRACe	Stores a specified trace to a specified waveform file.	5-50

Command	Function	Page
SENSe Group		
:SENSe:AVERage:COUNt	Sets or queries the number of times averaging for each measured point.	5-51
:SENSe:BANDwidth|:BWIDth[:RESolution]	Sets or queries the measurment resolution.	5-51
:SENSe:CHOPper	Sets or queries chopper mode.	5-51
:SENSe:CORRection:LEVel:SHIFt	Sets or queries the offset value for the level.	5-51
:SENSe:CORRection:RVELocity:MEDium	Sets or queries whether air or vacuum is used as the wavelength reference.	5-51
:SENSe:CORRection:WAVelength:SHIFt	Sets or queries the offset value for the levelwavelength.	5-51
:SENSe:SENSe:LEVel	Sets or queries the sensitivity based on the sensitivity level you want to measure at.	5-52
:SENSe:SENSe	Sets or queries the measurement sensitivity.	5-52
:SENSe:SETTing:CORRection	Sets or queries the resolution correction function.	5-52
:SENSe:SETTing:FIBer	Sets or queries the fiber core size mode.	5-52
:SENSe:SETTing:SMOothing	Sets or queries the Smoothing function.	5-52
:SENSe:SWEep:POINts	Sets or queries the number of samples measured.	5-52
:SENSe:SWEep:POINts:AUTO	Sets or queries the function of automatically setting the sampling number to be measured.	5-53
:SENSe:SWEep:SPEed	Sets or queries the sweep speed.	5-53
:SENSe:SWEep:STEP	Sets or queries the sampling interval for measurements.	5-53
:SENSe:SWEep:TIME:0NM	Sets or queries the time taken from the start to the end of measurements when measurement is made in the 0-nm sweep mode.	5-53
:SENSe:SWEep:TIME:INTerval	Sets or queries the time taken from the start of a sweep to that of the next sweep when repeat sweeps are made.	5-53
:SENSe:WAVelength:CENTer	Sets or queries the measurement condition center wavelength.	5-53
:SENSe:WAVelength:SPAN	Sets or queries the measurement condition measurement span.	5-54
:SENSe:WAVelength:SRANge	Sets or queries whether to limit a sweep range to the spacing between line markers L1 and L2.	5-54
:SENSe:WAVelength:STARt	Sets or queries the measurement condition measurement start wavelength.	5-54
:SENSe:WAVelength:STOP	Sets or queries the measurement condition measurement stop wavelength.	5-54
STATus Group		
:STATus:OPERation:CONDition?	Queries the contents of the operation status condition register.	5-55
:STATus:OPERation:ENABle	Queries the contents of the operation status Enable register.	5-55
:STATus:OPERation[:EVENt]?	Queries the contents of the operation status Event register.	5-55
:STATus:PRESet	Clears the event register and sets all bits of the enable register.	5-55
:STATus:QUEStionable:CONDition?	Queries the contents of the qestionable status condition register.	5-55
:STATus:QUEStionable:ENABle	Reads the contents of the questionable status enable register or writes data to this register.	5-55
:STATus:QUEStionable[:EVENt]?	Reads the contents of the questionable status event register.	5-55
SYSTem Group		
:SYSTem:BUZZer:CLICk	Sets or queries whether to sound the buzzer when clicked the key.	5-56
:SYSTem:BUZZer:WARNing	Sets or queries whether to sound the buzzer during an alarm.	5-56
:SYSTem:COMMunicate:CFORmat	Sets or queries the command format of this unit.	5-56
:SYSTem:COMMunicate:LOCKout	Sets/cancels local lockout.	5-57
:SYSTem:COMMunicate:RMONitor	Sets or queries whether the remote monitor function is enabled.	5-57
:SYSTem:DATE	Sets or queries the system data.	5-57
:SYSTem:DISPlay:UNCal	Sets or queries whether to display an alarm message in the event of UNCAL.	5-57
:SYSTem:ERRor[:NEXT]?	Queries data in an error queue and deletes it from the queue.	5-57
:SYSTem:GRID	Sets or queries the instrument's grid setting.	5-57
:SYSTem:GRID:CUSTom:SPACing	Sets or queries the grid spacing of the custom grid.	5-57
:SYSTem:GRID:CUSTom:STARt	Sets or queries the custom grid start wavelength.	5-57
:SYSTem:GRID:CUSTom:STOP	Sets or queries the custom grid stop wavelength.	5-58
:SYSTem:GRID:REFerence	Sets or queries the reference frequency of the instrument's grid setting.	5-58
:SYSTem:INFormation?	Queries model-specific information (the model code and special code)	5-58
:SYSTem:OLOCk	Sets or queries whether keys are locked.	5-58
:SYSTem:PRESet	Initializes the unit status.	5-58
:SYSTem:TIME	Sets or queries the system time.	5-58
:SYSTem:VERSion?	Queries the SCPI compatibility version of this unit.	5-58

Command	Function	Page
TRACe Group		
:TRACe:ACTive	Sets or queries the active trace.	5-59
:TRACe:ATTRibute[:<trace name>]	Sets or queries the attributes of the specified trace.	5-59
:TRACe:ATTRibute:RAVG[:<trace name>]	Sets or queries the number of times for averaging of the specified trace.	5-59
:TRACe:COPY	Copies the data of a specified trace to another trace.	5-59
:TRACe[:DATA]:SNUMber?	Sets or queries the number of number of data sampled of the specified trace.	5-59
:TRACe[:DATA]:X?	Queries the wavelength axis data of the specified trace.	5-60
:TRACe[:DATA]:Y?	Queries the level axis data of specified trace.	5-60
:TRACe[:DATA]:Y:PDENsity?	Queries the power spectral density trace data.	5-61
:TRACe:DELete	Deletes the data of a specified trace.	5-61
:TRACe:DELete:ALL	Clears the data for all traces.	5-61
:TRACe:STATe[:<trace name>]	Sets or queries the display status of the specified trace.	5-61
TRIGger Group		
:TRIGger[:SEQuence]:DELay	Sets or queries the trigger delay.	5-62
:TRIGger[:SEQuence]:GATE:ITIMe	Sets or queries sampling interval for gate sampling.	5-62
:TRIGger[:SEQuence]:GATE:LOGic	Sets or queries the gate signal logic of gate sampling.	5-62
:TRIGger[:SEQuence]:GATE:MDELay	Sets or queries the measure delay of gate sampling.	5-62
:TRIGger[:SEQuence]:SLOPe	Sets or queries the trigger edge.	5-62
:TRIGger[:SEQuence]:STATe	Sets or queries the external trigger mode.	5-62
:TRIGger[:SEQuence]:INPut	Sets or queries the signal of the input trigger.	5-62
:TRIGger[:SEQuence]:OUTPut	Sets or queries the signal of the output trigger.	5-63
:TRIGger[:SEQuence]:PHOLd:HTIMe	Sets or queries the hold time of peak hold mode.	5-63
UNIT Group		
:UNIT:POWer:DIGit	Sets or queries the number of decimal places displayed for the level value.	5-64
:UNIT:X	Sets or queries the units for the X axis.	5-64
Common Commands Group		
*CLS(Clear Status)	Clears all event status registers, the summary of which is reflected in the status byte register.	5-66
*ESE(Standard Event Status Enable)	Sets or queries the standard event enable register.	5-66
*ESR?(Standard Event Status Register)	Queries the standard event status register and simultaneously clears it.	5-66
*IDN? (Identification)	Queries the instrument type and firmware version.	5-66
*OPC(Operation Complete)	Sets or queries bit 0 (OPC) of the standard event status register (ESR) if operations waiting to be processed have all been completed.	5-66
*RST (Reset)	Executes a device reset to return the instrument to the known (default) status.	5-66
*SRE(Service Request Enable)	Sets or queries the service request enable register.	5-67
*STB?(Read Status Byte)	Queries the current value of the status byte register.	5-67
*TRG(Trigger)	Performs a <SINGLE> sweep under the sweep conditions established immediately before receiving the command.	5-67
*TST?(Self Test)	Performs the instrument's self-test and queries the status.	5-67
*WAI(Wait to Continue)	Prevents the instrument from executing another command until the execution of the current command is complete.	5-67

5.1 ABORt Group

:ABORt

Function Stops operations such as measurements and calibration.

Syntax `:ABORt`

Example `:ABORT`

Description
- Operations to be stopped are as follows:
  :APPLication:DLOGging:STATe
  :APPLication:PROGram:EXECute
  :CALibration:ALIGn[:IMMediate]
  :CALibration:ALIGn:INTernal[:IMMediate]
  :CALibration:WAVelength:EXTernal
  [:IMMediate]
  :CALibration:WAVelength:INTernal[:IMMediate]
  *TRG
  :INITiate
- This is an overlapping command.

5.2 APPLication Group

:APPLication:DLOGging:ETIMe?

Function Queries the elapsed time of data logging (in seconds).

Syntax `:APPLication:DLOGging:ETIMe?`
Response <integer>
<integer> = Elapsed time [sec]

Example `:APPLICATION:DLOGGING:ETIME? -> 10220`

Description
- This is an overlap command.
- This command is invalid when data logging is paused.

:APPLication:DLOGging:LPARameter:INTerval

Function Sets or queries the measurement interval of data logging.

Syntax `:APPLication:DLOGging:LPARameter:INTerval<wsp><integer>[SEC]`
`:APPLication:DLOGging:LPARameter:INTerval?`
<integer> = Measurement interval [sec]
(0 = SWEEP TIME)

Example `:APPLICATION:DLOGGING:LPARAMETER:INTERVAL 10`
`:APPLICATION:DLOGGING:LPARAMETER:INTERVAL? -> 10`

Description
- This command is invalid when data logging is in progress.
- This is a sequential command.

:APPLication:DLOGging:LPARameter:ITEM

Function Sets or queries the data logging source.

Syntax `:APPLication:DLOGging:LPARameter:ITEM<wsp>0|1|2|3`
`:APPLication:DLOGging:LPARameter:ITEM?`
0|1|2|3: Data logging source
0 = WDM, 1 = PEAK, 2 = MULTI-PEAK,
3 = DFB-LD

Example `:APPLICATION:DLOGGING:LPARAMETER:ITEM 0`
`:APPLICATION:DLOGGING:LPARAMETER:ITEM? -> 0`

Description
- This command is invalid when data logging is in progress.
- This is a sequential command.

:APPLication:DLOGging:LPARameter:LMODe

Function Sets or queries the data logging mode (maximum channel mode or maximum logging mode).

Syntax `:APPLication:DLOGging:LPARameter:LMODe<wsp>1|2`
`:APPLication:DLOGging:LPARameter:LMODe?`
1|2: Mode
1 = Maximum channel mode (MODE1: MAX 1024ch, 2001 entries)
2 = Maximum logging mode (MODE2: MAX 256ch, 10001 entries)

Example `:APPLICATION:DLOGGING:LPARAMETER:LMODE 1`
`:APPLICATION:DLOGGING:LPARAMETER:LMODE -> 1`

Description
- This command is invalid when data logging is in progress.
- This is a sequential command.

:APPLication:DLOGging:LPARameter:MEMory

Function Sets or queries the temporary area for saving waveform files of data logging.

Syntax `:APPLication:DLOGging:LPARameter:MEMory<wsp>INTernal|EXTernal`
`:APPLication:DLOGging:LPARameter:MEMory?`
INTernal = Internal memory
EXTernal = USB storage media

Example `:APPLICATION:DLOGGING:LPARAMETER:MEMORY INTERNAL`
`:APPLICATION:DLOGGING:LPARAMETER:MEMORY? -> INT`

Description
- This command is invalid when data logging is in progress.
- This is a sequential command.

:APPLication:DLOGging:LPARameter:MTHResh

Function Sets or queries the threshold of the channel-matching wavelength λ for data logging.

Syntax
```
:APPLication:DLOGging:LPARameter:MTHResh<wsp><NRf>[M]
:APPLication:DLOGging:LPARameter:MTHResh?
```
<NRf>[M] = Threshold of wavelength λ [m]

Example
```
:APPLICATION:DLOGGING:LPARAMETER:MTHResh 0.1nm
:APPLICATION:DLOGGING:LPARAMETER:MTHResh? -> +1.00000000E-010
```

Description
- This command is invalid when data logging is in progress.
- This is a sequential command.

:APPLication:DLOGging:LPARameter:PDETect:ATHResh

Function Sets or queries the threshold (absolute value) for detecting the data logging mode.

Syntax
```
:APPLication:DLOGging:LPARameter:PDETect:ATHResh<NRf>[DBM]
:APPLication:DLOGging:LPARameter:PDETect:ATHResh?
```
<NRf>[DBM] = Peak detection threshold (absolute value) [dBm]

Example
```
:APPLICATION:DLOGGING:LPARAMETER:PDETECT:ATHRESH -20.0dbm
:APPLICATION:DLOGGING:LPARAMETER:PDETECT:ATHRESH? -> -2.00000000E+001
```

Description
- This command is invalid when data logging is in progress.
- This is a sequential command.

:APPLication:DLOGging:LPARameter:PDETect:RTHResh

Function Sets or queries the threshold (relative value) for detecting the data logging mode.

Syntax
```
:APPLication:DLOGging:LPARameter:PDETect:RTHResh<NRf>[DB]
:APPLication:DLOGging:LPARameter:PDETect:RTHResh?
```
<NRf>[DB] = Peak detection threshold (relative value) [dB]

Example
```
:APPLICATION:DLOGGING:LPARAMETER:PDETECT:RTHRESH 30.0db
:APPLICATION:DLOGGING:LPARAMETER:PDETECT:RTHRESH? -> +3.00000000E+001
```

Description
- This command is invalid when data logging is in progress.
- This is a sequential command.

:APPLication:DLOGging:LPARameter:PDETect:TTYPe

Function Sets or queries how the threshold for detecting the data logging mode (peak or bottom) is specified.

Syntax
```
:APPLication:DLOGging:LPARameter:PDETect:TTYPe<wsp>ABSolute|RELative
:APPLication:DLOGging:LPARameter:PDETect:TTYPe?
```
ABSolute = Absolute value
RELative = Relative value

Example
```
:APPLICATION:DLOGGING:LPARAMETER:PDETECT:TTYPE ABSOLUTE
:APPLICATION:DLOGGING:LPARAMETER:PDETECT:TTYPE? -> ABS
```

Description
- This command is invalid when data logging is in progress.
- This is a sequential command.

:APPLication:DLOGging:LPARameter:TDURation

Function Sets or queries the measurement duration of data logging (in seconds).

Syntax
```
:APPLication:DLOGging:LPARameter:TDURation<wsp><integer>[sec]
:APPLication:DLOGging:LPARameter:TDURation?
```
<integer> = Measurement duration [sec]

Example
```
:APPLICATION:DLOGGING:LPARAMETER:TDURation 3600
:APPLICATION:DLOGGING:LPARAMETER:TDURation? -> 3600
```

Description
- This command is invalid when data logging is in progress.
- This is a sequential command.

:APPLication:DLOGging:LPARameter:TLOGging

Function Sets or queries whether waveforms will be logged during data logging.

Syntax
```
:APPLication:DLOGging:LPARameter:TLOGging<wsp>OFF|ON|0|1
:APPLication:DLOGging:LPARameter:TLOGging?
```
OFF = Waveform data save function off
ON = Waveform data save function on

Example
```
:APPLICATION:DLOGGING:LPARAMETER:TLOGGING OFF
:APPLICATION:DLOGGING:LPARAMETER:TLOGGING? -> 0
```

Description
- This command is invalid when data logging is in progress.
- This is a sequential command.

:APPLication:DLOGging:STATe

Function Starts, stops, or queries data logging.

Syntax `:APPLication:DLOGging:STATe<wsp>STOP|STARt|0|1`
`:APPLication:DLOGging:STATe?`
START = Starts data logging
STOP = Stops data logging
Response 0 = Stopped, 1 = Running

Example `:APPLICATION:DLOGGING:STATE 1`
`:APPLICATION:DLOGGING:STATE? -> 1`

Description
- Only the following commands are valid when data logging is in progress.
- Common commands (excluding *TRG and *TST)
- All query commands
- ABORt
- This is an overlappable command.

:APPLication:PROGram:EXECute

Function Executes a program that has been specified.

Syntax `:APPLication:PROGram:EXECute<wsp><integer>`
<integer> = Number of a program to execute

Example `:APPLICATION:PROGRAM:EXECUTE 1`

Explanation This is an overlapable command.

5.3 CALCulate Group

Commands about the following functions are summarized in this sub system.

- Analysis function (Spectrum Width, ANALYSIS1 , ANALYSIS2)
- Peak/Bottom search function
- Marker function (Δ marker, line marker)
- Calculation function of trace
- Advanced marker function (moving marker, power spectral density marker, integrated power marker)

The following procedure is performed in order to carry out remote control of the Analysis function.

1. Select the analysis algorithm (:CALCulate:CATegory command)
2. Set the Analysis Parameter (:CALCulate:PARameter command)
3. Execute the analysis function (:CALCulate[:IMMediate] command)
4. Get the analysis results (:CALCulate:DATA? command)

The following command is used in order to carry out remote control of the Peak/Bottom search function.

:CALCulate:MARKer:MAXimum|MINimum command

The following command is used to in order to carry out remote control of the Marker function.

Δ marker :CALCulate:MARKer command

Line marker :CALCulate:LMARker command

The following command is used to in order to carry out remote control of the trace Calculation function.

:CALCulate:MATH command

:CALCulate:ARESolution?

Function Queries the actual resolution data of the specified trace.

Syntax `:CALCulate:ARESolution?<wsp>`
`<trace name>,[<start point>,`
`<stop point>]`
<trace name> = Target trace (TRA|TRB|TRC|TRD|TRE|TRF|TRG)
<start point> = Sample range to transfer (start point) (1 to 200001)
<stop point> = Sample range to transfer (stop point) (1 to 200001)

Example
```
CALCULATE:ARESOLUTION? TRA
-> +1.89759145E-009,+1.89744762E-009,
+1.89730346E-009,....
```

Description
- The function outputs a wavelength value.
- If the <start point> and <stop point> parameters are omitted, the entire sample data of the specified trace will be output.
- The data is output in ASCII or BINARY format according to the :FORMat[:DATA] setting.
- This is a sequential command.

:CALCulate:CATegory

Function Sets or queries the type of analysis.

Syntax
```
:CALCulate:CATegory<wsp>{SWTHresh|
SWENvelope|SWRMs|SWPKrms|NOTCh|DFBLd|
FPLD|LED|SMSR|POWer|OSNR|WDM|NF|FILPk|
FILBtm|WFPeak|WFBTm|COLor|ITLa|
WDMSmsr0|1|2|3|4|5|6|7|8|9|11|12|13|14|15|
16|17|18|19}
:CALCulate:CATegory?
```

SWTHresh|0	Spectrum width analysis (THRESH)
SWENvelope|1	Spectrum width analysis (ENVELOPE)
SWRMs|2	Spectrum width analysis (RMS)
SWPKrms|3	Spectrum width analysis (PEAK-RMS)
NOTCh|4	Notch width analysis
DFBLd|5	DFB-LD parameter analysis
FPLD|6	FP-LD parameter analysis
LED|7	LED parameter analysis
SMSR|8	SMSR analysis
POWer|9	Power analysis
WDM|11	WDM analysis
NF|12	NF analysis
FILPk|13	Filter peak analysis
FILBtm|14	Filter bottom analysis
WFPeak|15	WDM FIL-PK analysis
WFBtm|16	WDM FIL-BTM analysis
COLor|17	COLOR analysis
ITLa|18	TLS analysis
WDMSmsr|19	WDM SMSR analysis

Example
```
:CALCULATE:CATegory SWTHresh
:CALCULATE:CATegory? -> 0
```

Description
- Even when this command is executed, no analysis is performed unless the :CALCulate[:IMMediate] command is executed.
- This is a sequential command.

:CALCulate:DATA?

Function Queries the analysis results.

Syntax
```
:CALCulate:DATA?
```

Example
```
:CALCULATE:DATA?
```

Description
- Queries the analysis results from the last time analysis was executed.
- If the analysis function has not been executed, a query error occurs.
- For a response example, see chapter appendix 3.
- This is a sequential command.

:CALCulate:DATA:CGAin?

Function Queries the gain value of the EDFA-NF analysis results.

Syntax
```
:CALCulate:DATA:CGAin?
```

Example
```
:CALCULATE:DATA:CGAin?
-> +1.00000000E+001,+1.00000000E+001
```

Description
- If the analysis function has not been executed, a query error occurs.
- “0” is returned if there is no relevant return value (such as if the analysis executed was not EDFA-NF analysis)
- The number of channels to be output can be acquired by the :CALCulate:DATA:NCHannels? command.
- Data is output in either ASCII or binary form, depending on the setting of :FORMat[:DATA].
- This is a sequential command.

:CALCulate:DATA:CNF?

Function Queries the NF value of the EDFA-NF analysis results.

Syntax
```
:CALCulate:DATA:CNF?
```

Example
```
:CALCULATE:DATA:CNF? ->
+1.00000000E+001,+1.00000000E+001
```

Description
- If :CALCulate[:IMMediate] has not been executed, a query error occurs.
- “0” is returned if there is no relevant return value (such as if the analysis executed was not EDFA-NF analysis)
- The number of channels to be output can be acquired by the :CALCulate:DATA:NCHannels? command.
- Data is output in either ASCII or binary form, depending on the setting of :FORMat[:DATA].
- This is a sequential command.

:CALCulate:DATA:COLor?

Function Queries the dominant wavelength of the measured light source and the x coordinate, y coordinate, z coordinate, color temperature, and deviation of the measured light source on the xy chromaticity diagram.

Syntax `:CALCulate:DATA:COLor?`

Example
```
:CALCulate:DATA:COLor? ->
+0.58237440E-006,+4.30500000E- 001,
+4.03000000E-001,+1.66500000E- 001,
+3.10300000E+003,+0.0132000 0E+000
```

Explanation
- This command returns analysis results that are not returned from the ":CALCulate:DATA?" command, such as the OSNR value. The following items are returned. The items are indicated with symbols. For the meanings of the symbols, see section 6.7. <dominant wl>,<x col>,<y col>,<z col>,<color temp>,<dev>
- "0" is returned if there is no relevant return value.
- This is a sequential command.

:CALCulate:DATA:CPOWers?

Function Queries the level value of the OSNR(WDM), WDM, EDFA-NF, WDM FIL-PK, or WDM FIL-BTM analysis results.

Syntax `:CALCulate:DATA:CPOWers?`

Example
```
:CALCULATE:DATA:CPOWERS? ->
+1.00000000E+001,+1.00000000E+001
```

Description
- If the analysis function has not been executed, a query error occurs.
- "0" is returned if there is no relevant return value.
- The number of channels to be output can be acquired by the :CALCulate:DATA:NCHannels? command.
- The value to be output depends on the analysis performed.
  OSNR(WDM): LEVEL or MEAS LEVEL
  EDFA-NF: INPUT LEVEL
  WDM FIL-PK: PEAK LEVEL (output even if SW is OFF)
  WDM FIL-BTM:PEAK LEVEL (output even if SW is OFF)
- Data is output in either ASCII or binary form, depending on the setting of :FORMat[:DATA].
- This is a sequential command.

:CALCulate:DATA:CSNR?

Function Queries the SNR value from the last time WDM analysis was executed.

Syntax `:CALCulate:DATA:CSNR?`

Example
```
:CALCULATE:DATA:CSNR? ->
+4.00000000E+001,+4.00000000E+001
```

Description
- If the analysis function has not been executed, a query error occurs.
- "0" is returned if there is no relevant return value (for example, if analysis made is other than WDM analysis).
- The number of channels to be output can be acquired by the :CALCulate:DATA:NCHannels? command.
- Data is output in either ASCII or binary form, depending on the setting of :FORMat[:DATA].
- This is a sequential command.

:CALCulate:DATA:CWAVelengths?

Function Queries the wavelength value of the OSNR(WDM), WDM, EDFA-NF, WDM FIL-PK, or WDM FIL-BTM analysis results.

Syntax `:CALCulate:DATA:CWAVelengths?`

Example
```
:CALCULATE:DATA:CWAVELENGTHS? ->
+1.55000000E-006,+1.56000000E-006
```

Description
- If the analysis function has not been executed, a query error occurs.
- "0" is returned if there is no relevant return value.
- The number of channels to be output can be acquired by the :CALCulate:DATA:NCHannels? command.
- The value to be output depends on the analysis performed.
  OSNR(WDM) : WAVELENGTH or MEAS WL
  WDM : WAVELENGTH or MEAS WL
  EDFA-NF: WAVELENGTH
  WDM FIL-PK: NOMINAL WAVELENGTH
  WDM FIL-BTM: NOMINAL WAVELENGTH
- Data is output in either ASCII or binary form, depending on the setting of :FORMat[:DATA].
- This is a sequential command.

:CALCulate:DATA:DFBLd?

Function Queries the DFB-LD analysis results.

Syntax :CALCulate:DATA:DFBLd?

Example :CALCULATE:DATA:DFBLD? ->

Description
- If the :CALCulate[:IMMediate] command has not been executed, a query error occurs.
- "0" is returned if there is no relevant return value (for example, if the analysis that was executed was not a DFB-LD analysis).
- This command returns analysis results that are not returned from the ":CALCulate:DATA?" command, such as the OSNR value. The following items are returned. The items are listed here as abbreviations. For the meaning of these abbreviations, see chapter appendix 3. <peak wl>,<peak lvl>,<center wl>,<spec wd>,<smsr(L)>,<smsr(R)>,<mode ofst(L)>,<mode ofst(R)>,<snr>,<power>,<rms>,<Krms>
- This is a sequential command.

:CALCulate:DATA:NCHannels?

Function Queries the number of channels of the OSNR(WDM), WDM, EDFA-NF, WDM FIL-PK, WDM FIL-BTM, or WDM SMSR analysis results.

Syntax :CALCulate:DATA:NCHannels?

Example :CALCULATE:DATA:NCHANNELS? -> 16

Description
- If the analysis function has not been executed, a query error occurs.
- "0" is returned if there is no relevant return value.
- The value is output as ASCII data, regardless of the setting of FORMat[:DATA].
- This is a sequential command.

:CALCulate:DATA:OSLope?

Function Queries the OUTPUT SLOPE value of the OSNR (WDM) and WDM analysis results.

Syntax :CALCulate:DATA:OSLope?
Response <NRf> = Output slope value [dB/nm] or [dB/THz]

Example :CALCULATE:DATA:OSLOPE? ->
+2.45352623E-001

Description
- A query error will occur if the analysis function is not implemented.
- "0" is returned if there is no relevant return value (for example, if the analysis that was executed was not a WDM analysis).
- Analysis results can be queried even if the output of the OUTPUT SLOPE value is set to OFF.
- ASCII data is returned regardless of the setting specified by the :FORMat:[DATA] command.
- This is a sequential command.

:CALCulate:DISPlay

Function Sets or queries the display format of analysis results.

Syntax :CALCulate:DISPlay<wsp>0|1|2|3|4
:CALCulate:DISPlay?
0: TRACE&TABLE
1: TABLE
2: TRACE
3: GRAPH&TABLE
4: GRAPH

Example :CALCULATE:DISPLAY 1
:CALCULATE:DISPLAY? -> 1

Description This is a sequential command.

:CALCulate:DISPlay:GRAPh:LMARKer:Y

Function Sets or queries the position of line marker Y1 or Y2 on the graph display of analysis results.

Syntax :CALCulate:DISPlay:GRAPh:LMARker:Y
<wsp>1|2,<NRf>[DB]
:CALCulate:DISPlay:GRAPh:LMARker:Y?
<wsp>1|2
1: Line marker Y1. 2: Line marker Y2.
<NRf>: Line marker position

Example :CALCULATE:DISPLAY:GRAPH:LMARKER:
Y 1,3.4
:CALCULATE:DISPLAY:GRAPH:LMARKER:Y? 1
-> +3.40000000E+000

Description
- This command is valid when the EDFA-NF analysis results are being displayed on a graph.
- This is a sequential command.

:CALCulate[:IMMediate]

Function Executes analysis. Queries the result of whether analysis has been performed.

Syntax :CALCulate[:IMMediate]
:CALCulate[:IMMediate]?
0: Not performed
1: Performed

Example :CALCULATE
:CALCULATE? -> 1

Description
- Analysis is performed according to the latest analysis settings.
- Analysis is performed on the following occasions:
- When :CALCulate[:IMMediate] command is executed.
- When :CALCulatePARameter: command is executed, or parameter settings changed
- This is a sequential command.

:CALCulate[:IMMediate]:AUTO

Function Sets or queries the automatic analysis function.

Syntax
```
:CALCulate[:IMMediate]:AUTO<wsp>
OFF|ON|0|1
:CALCulate[:IMMediate]:AUTO?
```
0: OFF
1: ON

Example
```
:CALCULATE:AUTO ON
:CALCULATE AUTO? -> 1
```

Description
- When the automatic analysis function is ON, automatically activates an analysis function that is active after a sweep has ended.
- This is a sequential command.

:CALCulate:LMARker:AOFF

Function Clears all line markers.

Syntax
```
:CALCulate:LMARker:AOFFExample
:CALCULATE:LMARKER:AOFF
```

Description This is a sequential command.

:CALCulate:LMARker:SRANge

Function Sets or queries whether to limit an analytical range to the spacing between line markers L1 and L2.

Syntax
```
:CALCulate:LMARker:SRANge<wsp>OFF|
ON|0|1
:CALCulate:LMARker:SRANge?
```
0: OFF
1: ON

Example
```
:CALCULATE:LMARKER:SRANGE ON
:CALCULATE:LMARKER:SRANGE? -> 1
```

Description This is a sequential command.

:CALCulate:LMARker:SSPan

Function Sets spacing between line markers L1 and L2 for span.

Syntax
```
:CALCulate:LMARker:SSPan
```

Example
```
:CALCULATE:LMARKER:SSPAN
```

Description This is a sequential command.

:CALCulate:LMARker:SZSPan

Function Sets spacing between line markers L1 and L2 for zoom span.

Syntax
```
:CALCulate:LMARker:SZSPan
```

Example
```
:CALCULATE:LMARKER:SZSPAN
```

Description This is a sequential command.

:CALCulate:LMARker:X

Function Sets or queries the position of line markers L1 and L2.

Syntax
```
:CALCulate:LMARker:X<wsp>1|2,<NRf>
[M|HZ]
:CALCulate:LMARker:X?<wsp>1|2
```
1, 2 = Line marker numbers
<NRf> = Position of a line marker
Response <NRf> [m|Hz]

Example
```
:CALCULATE:LMARKER:X 1,1550.000nm
:CALCULATE:LMARKER:X? 1 ->
+1.55000000E-006
```

Description
- If the specified line marker is not located, a query error occurs.
- This is a sequential command.

:CALCulate:LMARker:Y

Function Sets or queries the position of line markers L3 and L4.

Syntax
```
:CALCulate:LMARker:Y<wsp>3|4,<NRf>
[DBM|DB|%]
:CALCulate:LMARker:Y?<wsp>3|4
```
3, 4 = Line marker numbers
<NRf> = Position of a line marker

Example
```
:CALCULATE:LMARKER:Y 3,-10dBm
:CALCULATE:LMARKER:Y? 3 ->
-1.00000000E+001
```

Description
- If the specified line marker is not located, a query error occurs.
- This is a sequential command.

:CALCulate:MARKer:AOFF

Function Clears all markers.

Syntax
```
:CALCulate:MARKer:AOFF
```

Example
```
:CALCULATE:MARKER:AOFF
```

Description This is a sequential command.

:CALCulate:MARKer:AUTO

Function Sets or queries the auto search function.

Syntax
```
:CALCulate:MARKer:AUTO<wsp>
OFF|ON|0|1
:CALCulate:MARKer:AUTO?
```
0 = OFF
1 = ON

Example
```
:CALCULATE:MARKER:AUTO  ON
:CALCULATE:MARKER:AUTO? -> 1
```

Description
- When the auto search function is ON, this instrument automatically performs a peak/bottom search through an active trace after a sweep has ended.
- This is a sequential command.

:CALCulate:MARKer:FUNCtion:FORMat

Function Sets the format of a difference value displayed in the area marker and queries the format set.

Syntax
```
:CALCulate:MARKer:FUNCtion:FORMat<wsp>OFFSet|SPACing|0|1
:CALCulate:MARKer:FUNCtion:FORMat?
```
OFFSet = Displays the difference of each marker relative to the moving marker.
SPACing = Displays the difference of each marker relative to a neighboring marker.
Response 0 = OFFSet, 1 = SPACing

Example
```
:CALCULATE:MARKER:FUNCTION:
FORMAT SPACING
:CALCULATE:MARKER:FUNCTION:FORMAT?
-> 1
```

Description This is a sequential command.

:CALCulate:MARKer:FUNCtion:UPDate

Function Sets or queries ON/OFF of the automatic update function of fixed markers used when updating an active trace.

Syntax
```
:CALCulate:MARKer:FUNCtion:UPDate<wsp>OFF|ON|0|1
:CALCulate:MARKer:FUNCtion:UPDate?
```
Response 0 = OFF, 1 = ON

Example
```
:CALCULATE:MARKER:FUNCTION:
UPDATE ON
:CALCULATE:MARKER:FUNCTION:UPDATE?
-> 1
```

Description
- When the automatic update function is ON and the active trace is updated, the level positions of fixed markers automatically follow the waveform.
- This is a sequential command.

:CALCulate:MARKer:IRANge

Function Sets or queries the integration frequency range of the specified integral marker.

Syntax
```
:CALCulate:MARKer:IRANge<wsp>
<marker>,<NRf>[HZ]
:CALCulate:MARKer:IRANge?<wsp>
<marker>
```
<marker> = Marker number
<NRf> = Integral range in Hz.
The setting is valid only when the type is set to Integral.

Example
```
:CALCULATE:MARKER:IRANGE 1,40GHZ
:CALCULATE:MARKER:IRANGE? 1 ->
+4.00000000E+013
```

Description
- This command is valid only when the specified marker is an integral marker.
  To specify an integral marker, use the CALC:MARK:TYPE or :CALC:MARK command.
  An error will occur if the specified marker is not an integral marker.
- This is a sequential command.

:CALCulate:MARKer:MAXimum

Function Detects a peak and places the moving marker on that peak.

Syntax
```
:CALCulate:MARKer:MAXimum
```

Example
```
:CALCULATE:MARKER:MAXIMUM
```

Description This is a sequential command.

:CALCulate:MARKer:MAXimum:LEFT

Function Detects the nearest peak existing on the left side of the current position of the moving marker and places the moving marker on that peak.

Syntax
```
:CALCulate:MARKer:MAXimum:LEFT
```

Example
```
:CALCULATE:MARKER:MAXIMUM:LEFT
```

Description
- If the moving marker is OFF, an execution error occurs.
- This is a sequential command.

:CALCulate:MARKer:MAXimum:NEXT

Function Detects the highest peak that is below the level of the current position of the moving marker and places the moving marker on that peak.

Syntax
```
:CALCulate:MARKer:MAXimum:NEXT
```

Example
```
:CALCULATE:MARKER:MAXIMUM:NEXT
```

Description
- If the moving marker is OFF, an execution error occurs.
- This is a sequential command.

:CALCulate:MARKer:MAXimum:RIGHt

Function Detects the nearest peak existing on the right side of the current position of the moving marker and places the moving marker on that peak.

Syntax `:CALCulate:MARKer:MAXimum:RIGHt`

Example `:CALCULATE:MARKER:MAXIMUM:RIGHT`

Description
- If the moving marker is OFF, an execution error occurs.
- This is a sequential command.

:CALCulate:MARKer:MAXimum:SCENter

Function Detects the peak wavelength and sets it as the measurement center waveform.

Syntax `:CALCulate:MARKer:MAXimum:SCENter`

Example `:CALCULATE:MARKER:MAXIMUM:SCENTER`

Description This is a sequential command.

:CALCulate:MARKer:MAXimum:SCENter:AUTO

Function Sets or queries ON/OFF of the function to automatically detect the peak wavelength and set it as the measurement center wavelength.

Syntax
```
:CALCulate:MARKer:MAXimum:SCENter:AUTO<wsp>OFF|ON|0|1
:CALCulate:MARKer:MAXimum:SCENter:AUTO?
```
Response 0 = OFF, 1 = ON

Example
```
:CALCULATE:MARKER:MAXIMUM:SCENTER:AUTO ON
:CALCULATE:MARKER:MAXIMUM:SCENTER:AUTO? -> 1
```

Description
- When this function is ON, this instrument automatically detects the peak wavelength of an active trace wavelength each time a sweep has ended, and sets it as the measurement center wavelength.
- This is a sequential command.

:CALCulate:MARKer:MAXimum:SRLevel

Function Detects the peak level and sets it for the reference level.

Syntax `:CALCulate:MARKer:MAXimum:SRLevel`

Example `:CALCULATE:MARKER:MAXIMUM:SRLEVEL`

Description This is a sequential command.

:CALCulate:MARKer:MAXimum:SRLevel:AUTO

Function Sets or queries ON/OFF of the function to automatically detect the peak level and sets it as the reference level.

Syntax
```
:CALCulate:MARKer:MAXimum:SRLevel:AUTO<wsp>OFF|ON|0|1
:CALCulate:MARKer:MAXimum:SRLevel:AUTO?
```
Response 0 = OFF, 1 = ON

Example
```
:CALCULATE:MARKER:MAXIMUM:SRLEVEL:AUTO ON
:CALCULATE:MARKER:MAXIMUM:SRLEVEL:AUTO? -> 1
```

Description
- When this function is ON, the instrument automatically detects the peak level of an active trace wavelength each time a sweep has ended, and sets it as the reference level.
- This is a sequential command.

:CALCulate:MARKer:MAXimum:SZCenter

Function Detects the peak wavelength and sets it as the display center wavelength.

Syntax `:CALCulate:MARKer:MAXimum:SZCenter`

Example `:CALCULATE:MARKER:MAXIMUM:SZCENTER`

Description This is a sequential command.

:CALCulate:MARKer:MINimum

Function Detects the bottom and places the moving marker on that bottom.

Syntax `:CALCulate:MARKer:MINimum`

Example `:CALCULATE:MARKER:MINIMUM`

Description This is a sequential command.

:CALCulate:MARKer:MINimum:LEFT

Function Detects the nearest bottom existing on the left side of the current position of the moving marker and places the moving marker on that bottom.

Syntax `:CALCulate:MARKer:MINimum:LEFT`

Example `:CALCULATE:MARKER:MINIMUM:LEFT`

Description
- If the moving marker is OFF, an execution error occurs.
- This is a sequential command.

:CALCulate:MARKer:MINimum:NEXT

Function Detects the lowest bottom that is above the level of the current position of the moving marker and places the moving marker on that bottom.

Syntax `:CALCulate:MARKer:MINimum:NEXT`

Example `:CALCULATE:MARKER:MINIMUM:NEXT`

Description
- If the moving marker is OFF, an execution error occurs.
- This is a sequential command.

:CALCulate:MARKer:MINimum:RIGHt

Function Detects the nearest bottom existing on the right side of the current position of the moving marker and places the moving marker on that side.

Syntax :CALCulate:MARKer:MINimum:RIGHt

Example :CALCULATE:MARKER:MINIMUM:RIGHT

Description
- If the moving marker is OFF, an execution error occurs.
- This is a sequential command.

:CALCulate:MARKer:MSEarch

Function Sets or queries the type of the search function.

Syntax :CALCulate:MARKer:MSEarch<wsp>OFF|ON|0|1
:CALCulate:MARKer:MSEarch?
OFF|0: Sets the search function to single search.
ON|1: Sets the search function to multi search.
Response 0 = OFF, 1 = ON

Example :CALCULATE:MARKER:MSEARCH on
:CALCULATE:MARKER:MSEARCH? -> 1

Description
- The search is executed as soon as you set the search function.
- This is a sequential command.

:CALCulate:MARKer:MSEarch:SORT

Function Sets or queries the sort order of the multi search detection list.

Syntax :CALCulate:MARKer:MSEarch:SORT<wsp>WAVelength|LEVel|0|1
:CALCulate:MARKer:MSEarch:SORT?
WAVelength|0: Wavelengths are displayed in order starting from the shortest wavelength.
LEVel|1: For the peak search, levels are displayed in order starting from the highest level. For the bottom search, levels are displayed in order starting from the lowest level.
Response 0 = WAVelength, 1 = LEVel

Example :CALCULATE:MARKER:MSEARCH:SORT WAV
:CALCULATE:MARKER:MSEARCH:SORT? -> 0

Description This is a sequential command.

:CALCulate:MARKer:MSEarch:THResh

Function Sets or queries the multi search threshold.

Syntax :CALCulate:MARKer:MSEarch:THResh<wsp><NRf>[DB]
:CALCulate:MARKer:MSEarch:THResh?
<NRf>: Threshold (dB)

Example :CALCULATE:MARKER:MSEARCH:THRESH 50DB
:CALCULATE:MARKER:MSEARCH:THRESH? -> +5.00000000E+001

Description This is a sequential command.

:CALCulate:MARKer{:PDENsity|:NOISe}{:BWIDth|:BANDwidth}

Function Sets or queries the normalization bandwidth of the specified power spectral density marker.

Syntax :CALCulate:MARKer{:PDENsity|:NOISe}{:BWIDth|:BANDwidth}<wsp><NRf>[m]
:CALCulate:MARKer{:PDENsity|:NOISe}{:BWIDth|:BANDwidth}?
<NRf> = Normalization bandwidth of the DENSity marker in units of m.

Example :CALCULATE:MARKER:PDENSITY:BWIDTH 0.1NM
:CALCULATE:MARKER:PDENSITY:BWIDTH? -> +1.00000000E-013

:CALCulate:MARKer:SCENter

Function Sets the wavelength of the current moving marker as the measurement center waveform.

Syntax :CALCulate:MARKer:SCENter

Example :CALCULATE:MARKER:SCENTER

Description
- If the moving marker is OFF, an execution error occurs.
- This is a sequential command.

:CALCulate:MARKer:SRLevel

Function Sets the current level of the moving marker for the reference level.

Syntax :CALCulate:MARKer:SRLevel

Example :CALCULATE:MARKER:SRLEVEL

Description
- If the moving marker is OFF, an execution error occurs.
- This is a sequential command.

:CALCulate:MARKer[:STATe]

Function Specified marker is positioned or deleted in the position of the moving marker. Also, queries the status of the specified marker.

Syntax
```
:CALCulate:MARKer[:STATe]<wsp>
<marker>,OFF|ON||DENSity|INTegral|0|1|2
|3[,<integra range>]
:CALCulate:MARKer[:STATe]?<wsp>
<marker>
```
<marker>: Marker number
1|NORMal Normal marker
2|DENSity Power spectral density marker
3|INTegral Integral marker
<integra range> = <NRf>[Hz]
Setting is possible when the marker type is set to INTegral(3).
Response 0 = OFF, 1 = NORMal
2 = DENSity,
3 = INTegral,<integra range>

Example
```
:CALCULATE:MARKER:STATE 1,ON
:CALCULATE:MARKER:STATE 1 -> 1
```

Description
- When the moving marker is not active and an attempt is made to set a fixed marker, an execution error occurs.
- If moving marker is specified, it is placed in the center of measurment display.
- This is a sequential command.

:CALCulate:MARKer:SZCenter

Function Sets the current wavelength of the moving marker for the display center wavelength.

Syntax
```
:CALCulate:MARKer:SZCenter
```

Example
```
:CALCULATE:MARKER:SZCENTER
```

Description
- If the moving marker is OFF, an execution error occurs.
- This is a sequential command.

:CALCulate:MARKer:TYPE

Function Sets or queries the marker type.

Syntax
```
:CALCulate:MARKer:TYPE<wsp><marker>,
<type>
:CALCulate:MARKer:TYPE?<wsp><marker>
```
<marker>: Marker number
<type>: 1|NORMAL Normal marker
2|DENSity Power spectral density marker
3|INTegral Integral marker

Example
```
:CALCULATE:MARKER:TYPE 1,DENS
:CALCULATE:MARKER:TYPE? 1 -> 2
```

Description This is a sequential command.

:CALCulate:MARKer:UNIT

Function Sets or queries the units of display for the marker values.

Syntax
```
:CALCulate:MARKer:UNIT<wsp>WAVelength
|FREQuency|WNUMber|0|1|2
:CALCulate:MARKer:UNIT?
```
Parameter
WAVelength|0
FREQuency|1
WNUMber|2
Response 0=WAVelength, 1= FREQuency,
2= WNUMber

Example
```
:CALCULATE:MARKER:UNIT FREQUENCY
:CALCULATE:MARKER:UNIT? -> 1
```

Description This is a sequential command.

:CALCulate:MARKer:X

Function Sets the specified marker at the specified position.Queries the X value of the specified marker.

Syntax
```
:CALCulate:MARKer:TYPE<wsp><marker>,
<type>
:CALCulate:MARKer:TYPE?<wsp><marker>
```
<marker>: Marker number (0 = Moving marker)
<type>: 1|NORMAL Normal marker
2|DENSity Power spectral density marker
3|INTegral Integral marker
If omitted, the Normal marker is set.
<integra range> = <NRf>[Hz]
Setting is possible when the marker type is set to INTegral(3).
Response If <marker> is specified
<NRf>[m|Hz|m^{-1}]]
If ALL is specified
<integer>, <NRf>, <NRf>, ..., <NRf>

Example
```
:CALCULATE:MARKER:TYPE 1,DENS
:CALCULATE:MARKER:TYPE? 1 -> 2
```

Description
- If an already located marker is specified, that marker will be moved to a specified position.
- If the specified marker is not located, a query error occurs.
- If ALL is specified (e.g., :CALC:MARK:X? ALL), the X values of all assigned markers will be returned.
- This is a sequential command.

:CALCulate:MARKer:Y?

Function Queries the Y value of the specified marker.

Syntax
```
:CALCulate:MARKer:Y?<wsp><marker>|
ALL
```
<marker> : Marker number (0: moveing marker)
ALL = All assigned markers
If <marker> is specified
<NRf>= Marker level
If ALL is specified
<integer>, <NRf>, <NRf>, ..., <NRf>

Example
```
:CALCULATE:MARKER:Y? 0
-> -1.00000000E+001
```

Description
- This unit of the marker level to be queried is dependent on the Y-axis unit of the active trace.
- If the specified marker is not located, a query error occurs.
- If ALL is specified (e.g., :CALC:MARK:Y? ALL), the Y values of all assigned markers will be returned.
- This is a sequential command.

:CALCulate:MATH:TRC

Function Sets or queries the TRACE C calculation function.

Syntax
```
:CALCulate:MATH:TRC<wsp>A-B(LOG)|
B-A(LOG)|A+B(LOG)|A+B(LIN)|
A-B(LIN)|B-A(LIN)|1-K(A/B)|
1-K(B/A)|
:CALCulate:MATH:TRC?
```

Example
```
:CALCULATE:MATH:TRC A-B(LOG)
:CALCULATE:MATH:TRC? -> A-B(LOG)
```

Description
- When the calculation function of trace C is set using this command, the attribute of trace C automatically becomes attribute “CALC”.
- If trace C is not a calculation trace, “NONE” is returned.
- This is a sequential command.

:CALCulate:MATH:TRC:K

Function Sets or queries parameter K of the TRACE C calculation function.

Syntax
```
:CALCulate:MATH:TRC:K<wsp><NRf>
:CALCulate:MATH:TRC:K?
```
<NRf> = Parameter K

Example
```
:CALCULATE:MATH:TRC:K 0.1
:CALCULATE:MATH:TRC:K?
-> +1.00000000E-001
```

Description This is a sequential command.

:CALCulate:MATH:TRF

Function Sets or queries the TRACE F calculation function.

Syntax
```
:CALCulate:MATH:TRF<wsp>C-D(LOG)|
D-C(LOG)|C+D(LOG)|D-E(LOG)|
E-D(LOG)|D+E(LOG)|C+D(LIN)|
C-D(LIN)|D-C(LIN)|D+E(LIN)|
D-E(LIN)|E-D(LIN)|PWRNBWA|PWRNBWB|
PWRNBWC|PWRNBWD|PWRNBWE
:CALCulate:MATH:TRF?
```

Example
```
:CALCULATE:MATH:TRF C-D(LOG)
:CALCULATE:MATH:TRF? -> C-D(LOG)
```

Description
- When the calculation function of trace F is set using this command, the attribute of trace F automatically becomes attribute “CALC”.
- If trace F is not a calculation trace, “NONE” is returned.
  Example :calc:math:trf c-d(log)
  :calc:math:trf? -> C-D(LOG)
- This is a sequential command.

:CALCulate:MATH:TRF:PNBW:BWIDth|BANDwidth

Function Sets or queries the normalization bandwidth of the power spectral density trace.

Syntax
```
:CALCulate:MATH:TRF:PNBW:BWIDth|
BANDwidth<wsp><NRf>[m]
:CALCulate:MATH:TRF:PNBW:BWIDth|
BANDwidth?
<NRf>=Normalization bandwidth[mm]
```

Example
```
:CALCULATE:MATH:TRF:PNBW:BAND 0.1nm
:CALCULATE:MATH:TRF:PNBW:BAND?
-> 1.00000000E-010
```

Description This is a sequential command.

:CALCulate:MATH:TRG

Function Sets or queries the TRACE G calculation function.

Syntax
```
:CALCulate:MATH:TRG<wsp>C-F(LOG)|
F-C(LOG)|C+F(LOG)|E-F(LOG)|
F-E(LOG)|E+F(LOG)|C+F(LIN)|
C-F(LIN)|F-C(LIN)|E+F(LIN)|
E-F(LIN)|FLIN)|NORMA|NORMB|NORMC|
CVFTA|CVFTB|CVFTC|MKRFT|PKCVFTA|
PKCVFTB|PKCVFTC
:CALCulate:MATH:TRG?
```

Example
```
:CALCULATE:MATH:TRG C-F(LOG)
:CALCULATE:MATH:TRG? -> C-F(LOG)
```

Description
- When the calculation function of trace G is set using this command, the attribute of trace G automatically becomes attribute “CALC”.
- If trace G is not a calculation trace, “NONE” is returned.
- This is a sequential command.

:CALCulate:MATH:TRG:CVFT:FALGo

Function Sets or queries the fitting curve function of the TRACE G fitting curve function.

Syntax
```
:CALCulate:MATH:TRG:CVFT:FALGo
<wsp><algorithm>
:CALCulate:MATH:TRG:CVFT:FALGo?
```
<algorithm>
GAUSs = GAUSS
LORENz = LORENZ
3RD = 3RD POLY
4TH = 4TH POLY
5TH = 5TH POLY
Response
0 = GAUSS 1 = LORENZ,
2 = 3RD POLY 3 = 4TH POLY
4 = 5TH POLY

Example
```
:CALCULATE:MATH:TRG:CVFT:
FALGO GAUSS
:CALCULATE:MATH:TRG:CVFT:FALG? -> 0
```

Description
- Setting of calculation area is common to curve fit and peak curve fit.
- This is a sequential command.

:CALCulate:MATH:TRG:CVFT:OPARea

Function Sets or queries a calculation area during curve fit and peak curve fit.

Syntax
```
:CALCulate:MATH:TRG:CVFT:OPARea
<wsp>ALL|INL1-L2|OUTL1-L2|0|1|2
:CALCulate:MATH:TRG:CVFT:OPARea?
```
ALL = all of the set wavelength range
INL1-L2 = range surrounding line marker 1 and 2
OUTL1-L2 = range outisde line markers 1 and 2
Response 0 = ALL, 1 = INL1-L2, 2 = OUTL1-L2

Example
```
:CALCULATE:MATH:TRG:CVFT:
OPAREA INl1-l2
:CALCULATE:MATH:TRG:CVFT:OPAREA?-> 1
```

Description
- Setting of calculation area is common to curve fit and peak curve fit.
- This is a sequential command.

:CALCulate:MATH:TRG:CVFT:THResh

Function Sets or queries the threshold value for curve fitting.

Syntax
```
:CALCulate:MATH:TRG:CVFT:THResh
<wsp><integer>[DB]
:CALCulate:MATH:TRG:CVFT:THResh?
```
<NRf> = Threshold level [dB]

Example
```
:CALCULATE:MATH:TRG:CVFT:THRESH 10db
:CALCULATE:MATH:TRG:CVFT:THRESH?
-> +1.00000000E+001
```

Description This is a sequential command.

:CALCulate:MATH:TRG:PCVFt:THResh

Function Sets or queries the threshold value for peak curve fitting.

Syntax
```
:CALCulate:MATH:TRG:PCVFt:THResh
<wsp><integer>[DB]
:CALCulate:MATH:TRG:PCVFt:THResh?
```
<NRf> = Threshold level [dB]

Example
```
:CALCULATE:MATH:TRG:PCVFT:
THRESH 10db
:CALCULATE:MATH:TRG:PCVFT:THRESH?
-> +1.00000000E+001
```

Description This is a sequential command.

:CALCulate:PARameter[:CATegory]:DFBLd

Function Sets or queries parameters for the DFB-LD analysis function.

Syntax
```
:CALCulate:PARameter[:CATegory]:DFBLd
<wsp><item>,<paramater>,<data>
:CALCulate:PARameter[:CATegory]:DFBLd
?<wsp><item>,<paramater>
```
<item> = Analytical item that sets parameter(s)
<parameter> = Parameter to be set
<data> = Setting data

<item>	<paramater>	<data>
SWIDth	ALGO	ENVelope|THResh|RMS|PKRMs
	TH	<NRf>[DB]
	TH2	<NRf>[DB]
	K	<NRf>
	MFIT	OFF|ON|0|1
	MDIFf	<NRf> [DB]
SMSR	SMODe	SMSR1|SMSR2|SMSR3|SMSR4
	SMASk	<NRf>[M]
	MDIFf	<NRf>[DB]
	SMPower	TDATa|NORMalized
	BWIDth	<NRf>[M]
RMS	ALGO	RMS|PKRMs
	TH	<NRf>[DB]
	K	<NRf>
	MDIFf	<NRf> [DB]
POWer	SPAN	<NRf>[M]
OSNR	MDIFf	<NRf>[DB]
	NALGo	AFIX|MFIX|ACENter|MCENter|PIT|0|1|2|3|4
	NAREa	<NRf>[M]
	MAREa	<NRf>[M]
	FALGo	LINear|GAUSs|LORenz|3RD|4TH|5TH|0|1|2|3|4|5
	NBW	<NRf>[M]
	SPOWer	PEAK|INTegral|0|1
	IRANge	<NRf>

Example
```
:CALCULATE:PARAMETER:
DFBLD SWIDTH,ALGO,THRESH
:CALCULATE:PARAMETER:DFBLD? SWIDTH,
ALGO -> THR
:CALCULATE:PARAMETER:DFBLD SMSR,
SMASK,0.5NM
:CALCULATE:PARAMETER:DFBLD?
SMSR,SMASK -> +5.00000000E-010
```

Description
- If a non-existing parameter is used for a combination, an execution error occurs. (such as combinations of SWIDth and SMODe)
- This is a sequential command.

:CALCulate:PARameter[:CATegory]:FILBtm

Function Sets or queries parameters for the FILTER-BTM analysis function.

Syntax
```
:CALCulate:PARameter[:CATegory]:
FILBtm<wsp><item>,<paramater>,<data>
:CALCulate:PARameter[:CATegory]:
FILBtm?<wsp><item>,<paramater>
```
<item> = Analytical item that sets parameter(s)
<parameter> = Parameter to be set
<data> = Data to be set

<item>	<paramater>	<data>
BLEVel	SW	OFF|ON|0|1
BWAVelength	SW	OFF|ON|0|1
CWAVelength	SW	OFF|ON|0|1
	ALGO	PEAK|BOTTom
	TH	<NRf>[DB]
	MDIFf	<NRf>[DB]
NWIDth	SW	OFF|ON|0|1
	ALGO	PEAK|BOTTom
	TH	<NRf>[DB]
	MDIFf	<NRf>[DB]
XTALk	SW	OFF|ON|0|1
	ALGO	PEAK|BOTTom|BLEVel|GRID
	TH	<NRf>[DB]
	MDIFf	<NRf>[DB]
	CSPace	<NRf>[M]
	SAREa	<NRf>[M]

Example
```
:CALCULATE:PARAMETER:
FILBTM CWAVELENGTH,ALGO,BOTTOM
:CALCULATE:PARAMETER:
FILBTM?CWAVELENGTH,ALGO -> BOTT
:CALCULATE:PARAMETER:FILBTM XTALK,
CSPACE,0.2NM
:CALCULATE:PARAMETER:FILBTM?
XTALK,CSPACEe -> +2.00000000E-010
```

Description
- If a non-existing parameter is used for a combination, an execution error occurs (a combination of CWAVelength and SAREa, etc.).
- This is a sequential command.

:CALCulate:PARameter[:CATegory]:FILPk

Function Sets or queries parameters for the FILTER PEAK analysis function.

Syntax
```
:CALCulate:PARameter[:CATegory]:FILPk
<wsp><item>,<paramater>,<data>
:CALCulate:PARameter[:CATegory]:
FILPk?<wsp><item>,<paramater>
```
<item> = Analytical item that sets parameter(s)
<parameter> = Parameter to be set
<data> = Data to be set

<item>	<paramater>	<data>
PLEVel	SW	OFF|ON|0|1
PWAVelength	SW	OFF|ON|0|1
MWAVelength	SW	OFF|ON|0|1
	ALGO	THResh|RMS
	TH	<NRf> [DB]
	K	<NRf>
	MFIT	OFF|ON|0|1
	MDIFf	<NRf> [DB]
SWIDth	SW	OFF|ON|0|1
	ALGO	THResh|RMS
	TH	<NRf> [DB]
	K	<NRf>
	MFIT	OFF|ON|0|1
	MDIFf	<NRf> [DB]
XTALk	SW	OFF|ON|0|1
	ALGO	THResh|PLEVel|GRID
	TH	<NRf> [DB]
	K	<NRf>
	MFIT	OFF|ON|0|1
	MDIFf	<NRf> [DB]
	CSPace	<NRf> [M]
	SAREa	<NRf> [M]
RWIDth	SW	OFF|ON|0|1
	TH	<NRf> [DB]
	MDIFf	<NRf> [DB]

Example
```
:CALCULATE:PARAMETER:FILPK SWIDTH,
ALGO,THRESH
:CALCULATE:PARAMETER:FILPK? SWIDTH,
ALGO -> THR
:CALCULATE:PARAMETER:FILPK XTALK,
CSPACE,0.5NM  :CALCULATE:PARAMETER:
FILPK? XTALK,CSPACE
-> +5.00000000E-010
```

Description
- If a non-existing parameter is used for a combination, an execution error occurs (a combination of SWIDth and CSPace, etc.).
- This is a sequential command.

:CALCulate:PARameter[:CATegory]:FPLD

Function Sets or queries parameters for the FP-LD analysis function.

Syntax
```
:CALCulate:PARameter[:CATegory]:FPLD
<wsp><item>,<paramater>,<data>
:CALCulate:PARameter[:CATegory]:FPLD?
<wsp><item>,<paramater>
```
<item> = Analytical item that sets parameter(s)
<parameter> = Parameter to be set
<data> = Setting data

<item>	<paramater>	<data>
SWIDth	ALGO	ENVelope|THResh|RMS|PKRMs
	TH	<NRf> [DB]
	TH2	<NRf> [DB]
	K	<NRf>
	MFIT	OFF|ON|0|1
	MDIFf	<NRf> [DB]
MWAVelength	ALGO	ENVelope|THResh|RMS|PKRMs
	TH	<NRf> [DB]
	TH2	<NRf> [DB]
	K	<NRf>
	MFIT	OFF|ON|0|1
	MDIFf	<NRf> [DB]
TPOWer	OFFSet	<NRf> [DB]
MNUMber	ALGO	ENVelope|THResh|RMS|PKRMs
	TH	<NRf> [DB]
	TH2	<NRf> [DB]
	K	<NRf>
	MFIT	OFF|ON|0|11
	MDIFf	<NRf> [DB]

Example
```
:CALCULATE:PARAMETER:FPLD SWIDTH,
ALGO,THRESH
:CALCULATE:PARAMETER:FPLD? SWIDTH,
ALGO -> THR
:CALCULATE:PARAMETER:FPLD TPOWER,
OFFSET,1.0DB
:CALCULATE:PARAMETER:FPLD? TPOWER,
OFFSET -> +1.00000000E+000
```

Description
- If a non-existing parameter is used for a combination, an execution error occurs. (a combination of SWIDth and OFFSET, etc.)
- This is a sequential command.

:CALCulate:PARameter[:CATegory]:ITLa

Function Sets or queries parameters for the TLS analysis function.

Syntax
```
:CALCulate:PARameter[:CATegory]:ITLa
<wsp><item>,<paramater>,<data>
:CALCulate:PARameter[:CATegory]:ITLa?
<wsp><item>,<paramater>
```
<item> = Analytical item that sets parameter(s)
<parameter> = Parameter to be set
<data> = Setting data

<item>	<paramater>	<data>
SWIDth	ALGO	ENVelope|THResh|RMS|PKRMs
	TH	<NRf>[DB]
	TH2	<NRf>[DB]
	K	<NRf>
	MFIT	OFF|ON|0|1
	MDIFf	<NRf> [DB]
SMSR	SMODe	SMSR1|SMSR2|SMSR3|SMSR4
	SMASk	<NRf>[M]
	MDIFf	<NRf>[DB]
	SMPower	TDATa|NORMalized
	BWIDth	<NRf>[M]
POWer	SPAN	<NRf>[M]
SSER	ALGO	IEC|CURVefit
	MDIFf	<NRf>[DB]
	AARea	<NRf>[M]
	FARea	<NRf>[M]
	MARea	<NRf>[M]
	FALGo	GAUSs|LORenz|3RD|4TH|5TH|1|2|3|4|5
	NBW	<NRf>[M]
	SPOWer	PEAK|INTegral|0|1
	IRANge	<NRf>

Example
```
:CALCULATE:PARAMETER:ITLA SWIDTH,
ALGO,THRESH
:CALCULATE:PARAMETER:ITLA?
SWIDTH,ALGO -> THR
:CALCULATE:PARAMETER:ITLA SWIDTH,
TH, 20.00DB
:CALCULATE:PARAMETER:ITLA? SWIDTH,
TH -> +2.00000000E+000
```

Description This is a sequential command.

:CALCulate:PARameter[:CATegory]:LED

Function Sets or queries parameters for the LED analysis function.

Syntax
```
:CALCulate:PARameter[:CATegory]:LED
<wsp><item>,<paramater>,<data>
:CALCulate:PARameter[:CATegory]:LED?
<wsp><item>,<paramater>
```
<item> = Analytical item that sets parameter(s)
<parameter> = Parameter to be set
<data> = Setting data

<item>	<paramater>	<data>
SWIDth	ALGO	ENVelope|THResh|RMS|PKRMs
	TH	<NRf> [DB]
	TH2	<NRf> [DB]
	K	<NRf>
	MFIT	OFF|ON|0|1
	MDIFf	<NRf> [DB]
MWAVelength	ALGO	ENVelope|THResh|RMS|PKRMs
	TH	<NRf> [DB]
	TH2	<NRf> [DB]
	K	<NRf>
	MFIT	OFF|ON|0|1
	MDIFf	<NRf> [DB]
TPOWer	OFFSet	<NRf> [DB]

Example
```
:CALCULATE:PARAMETER:LED SWIDTH,ALGO,
THRESH
:CALCULATE:PARAMETER:LED? SWIDTH,ALGO
-> THR
:CALCULATE:PARAMETER:LED TPOWER,
OFFSET,1.0DB  :CALCULATE:PARAMETER:
LED? TPOWER,OFFSET -> +1.00000000E+000
```

Description
- If a non-existing parameter is used for a combination, an execution error occurs (a combination of SWIDth and OFFSet, etc.).
- This is a sequential command.

:CALCulate:PARameter[:CATegory]:NF:AALGo

Function Sets or queries the measurement algorithm applied to ASE level measurements made by the NF analysis function.

Syntax
```
:CALCulate:PARameter[:CATegory]:NF:AALGo<wsp><algorithm>
:CALCulate:PARameter[:CATegory]:NF:AALGo?
```
<algorithm> = Measurement algorithm
- AFIX: AUTO FIX
- MFIX: MANUAL FIX
- ACENter: AUTO CENTER
- MCENter: MANUAL CENTER

Response 0 = AUTO FIX
- 1 = MANUAL FIX
- 2 = AUTO CENTER
- 3 = MANUAL CENTER

Example
```
:CALCULATE:PARAMETER:NF:AALGO MFIX
:CALCULATE:PARAMETER:NF:AALGO? -> 1
```
Description This is a sequential command.

:CALCulate:PARameter[:CATegory]:NF:FALGo

Function Sets or queries the fitting function during level measurement applied to ASE level measurements made by the NF analysis function.

Syntax
```
:CALCulate:PARameter[:CATegory]:NF:FALGo<wsp><algorithm>
:CALCulate:PARameter[:CATegory]:NF:FALGo?
```
<algorithm> = Fitting function
- LINear: LINEAR
- GAUSs: GAUSS
- LORenz: LORENZ
- 3RD: 3RD POLY
- 4TH: 4TH POLY
- 5TH: 5TH POLY

Response 0 =LINEAR
- 1 = GAUSS
- 2 = LORENZ
- 3 = 3RD POLY
- 4 = 4TH POLY
- 5 = 5TH POLY

Example
```
:CALCULATE:PARAMETER:NF:FALGO GAUSS
:CALCULATE:PARAMETER:NF:FALGO? -> 1
```
Description This is a sequential command.

:CALCulate:PARameter[:CATegory]:NF:FAREa

Function Sets or queries the fitting range for level measurement applied to ASE level measurements made by the NF analysis function.

Syntax
```
:CALCulate:PARameter[:CATegory]:NF:FAREa<wsp><NRf>[M]
:CALCulate:PARameter[:CATegory]:NF:FAREa?
```
<NRf>= fitting range [m]

Example
```
:CALCULATE:PARAMETER:NF:FAREA 0.80NM
:CALCULATE:PARAMETER:NF:FAREA?
-> +8.00000000E-010
```
Description
- When the fitting range is set to “Between CH” (and ASE measurement algorithm is set to “AUTO-CTR” or “MANUAL-CTR”), then the command returns 0.
- This is a sequential command.

:CALCulate:PARameter[:CATegory]:NF:IOFFset

Function Sets or queries level offset values (signal light) for the NF analysis function.

Syntax
```
:CALCulate:PARameter[:CATegory]:NF:IOFFset<wsp><NRf>[DB]
:CALCulate:PARameter[:CATegory]:NF:IOFFset?
```
<NRf> = Level offset value of signal light [dB]

Example
```
:CALCULATE:PARAMETER:NF:IOFFSET 10.00
:CALCULATE:PARAMETER:NF:IOFFSET?
-> +1.00000000E+001
```
Description This is a sequential command.

:CALCulate:PARameter[:CATegory]:NF:IRANge

Function Sets or queries the integration frequency range for when the EDFA-NF analysis feature calculates the signal optical power.

Syntax
```
:CALCulate:PARameter[:CATegory]:NF:IRANge<wsp><NRf>
:CALCulate:PARameter[:CATegory]:NF:IRANge?
```
<NRf> = Integration range [GHz]

Example
```
:CALCulate:PARameter:NF:IRANGE 40
:CALCulate:PARameter:NF:IRANGE?
-> +4.00000000E+001
```
Description This is a sequential command.

:CALCulate:PARameter[:CATegory]:NF:MAREa

Function Sets or queries the mask range for level measurement applied to ASE level measurements made by the NF analysis function.

Syntax
```
:CALCulate:PARameter[:CATegory]:NF:
MAREa<wsp><NRf>[M]
:CALCulate:PARameter[:CATegory]:NF:
MAREa?
```
<NRf> = mask range [m]

Example
```
:CALCULATE:PARAMETER:NF:
MAREA 0.40NM
:CALCULATE:PARAMETER:NF:MAREA?
-> +4.00000000E-010
```

Description
- When the mask range is set to "---" (and ASE level measurement function is set to "LINEAR"), the command returns 0.
- This is a sequential command.

:CALCulate:PARameter[:CATegory]:NF:MDIFf

Function Sets or queries the peak bottom difference of channel detection for the NF analysis function.

Syntax
```
:CALCulate:PARameter[:CATegory]:NF:
MDIFf<wsp><NRf>[DB]
:CALCulate:PARameter[:CATegory]:NF:
MDIFf?
```
<NRf> = Peak bottom difference [dB]

Example
```
:CALCULATE:PARAMETER:NF:
MDIFF 3.00DB
:CALCULATE:PARAMETER:NF:MDIFF?
-> +3.00000000E+000
```

Description This is a sequential command.

:CALCulate:PARameter[:CATegory]:NF:OOFFset

Function Sets or queries level offset values (output light) for the NF analysis function.

Syntax
```
:CALCulate:PARameter[:CATegory]:NF:
OOFFset<wsp><NRf>[DB]
:CALCulate:PARameter[:CATegory]:NF:
OOFFset?
```
<NRf> = Level offset value of output light [dB]

Example
```
:CALCULATE:PARAMETER:NF:
OOFFSET 10.00
:CALCULATE:PARAMETER:NF:OOFFSET?
-> +1.00000000E+001
```

Description This is a sequential command.

:CALCulate:PARameter[:CATegory]:NF:PDISplay

Function Sets or queries whether to display data used for fitting of the NF analysis function on the waveform screen.

Syntax
```
:CALCulate:PARameter[:CATegory]:NF:PDI
Splay<wsp>OFF|ON|0|1
:CALCulate:PARameter[:CATegory]:NF:PDI
Splay?
```
Response 0 = OFF, 1 = ON

Example
```
:CALCULATE:PARAMETER:NF:PDISPLAY ON
:CALCULATE:PARAMETER:NF:PDISPLAY?-> 1
```

Description
- When this set value is 1 (ON), data used for fitting is displayed on the waveform screen.
- This is a sequential command.

:CALCulate:PARameter[:CATegory]:NF:TH

Function Sets or queries the threshold level of channel detection for the NF analysis function.

Syntax
```
:CALCulate:PARameter[:CATegory]:NF:
TH<wsp><NRf>[DB]
:CALCulate:PARameter[:CATegory]:NF:
TH?
```
<NRf> = Threshold level [dB]

Example
```
:CALCULATE:PARAMETER:NF:TH 20.00DB
:CALCULATE:PARAMETER:NF:TH->
+2.00000000E+001
```

Description This is a sequential command.

:CALCulate:PARameter[:CATegory]:NF:RBWidth

Function Sets or queries the method for calculating the resolution value of the NF computation.

Syntax
```
:CALCulate:PARameter[:CATegory]:NF:
RBWidth<wsp>MEASured|CAL|0|1
:CALCulate:PARameter[:CATegory]:NF:
RBWidth?
```
MEASured|0 Use the value determined from the waveform using THRESH 3dB analysis.

CAL|1 Use the actual resolution value stored in the instrument .

Response 0=MEASURED, 1=CAL

Example
```
:CALCULATE:PARAMETER:NF:RBWIDTH
MEASURED
:CALCULATE:PARAMETER:NF:RBWIDTH?
-> 0
```

Description This is a sequential command.

:CALCulate:PARameter[:CATegory]:NF:SNOise

Function Sets or queries whether Shot Noise is included in the NF computation

Syntax
```
:CALCulate:PARameter[:CATegory]:NF:
SNOise<wsp>OFF|ON|0|1
:CALCulate:PARameter[:CATegory]:NF:
SNOise?
```
OFF|0 Shot Noise not included in the NF computation
ON|1 Shot Noise included in the NF computation
Response 0=OFF, 1=ON

Example
```
:CALCULATE:PARAMETER:NF:SNOISE OFF
:CALCULATE:PARAMETER:NF:SNOISE?-> 0
```

Description This is a sequential command.

:CALCulate:PARameter[:CATegory]:NF:SPOWer

Function Sets or queries the signal optical power calculation method of the EDFA-NF analysis feature.

Syntax
```
:CALCulate:PARameter[:CATegory]:NF:
SPOWer<wsp>PEAK|INTegral|0|1
:CALCulate:PARameter[:CATegory]:NF:
SPOWer?
```
PEAK|0: The signal optical power is set to the level of the mode peak.
INTegral|1: The signal optical power is set to the power obtained by integrating the spectrum.

Example
```
:CALCulate:PARameter:NF:SPOWer PEAK
:CALCulate:PARameter:NF:SPOWer?
-> 0
```

Description This is a sequential command.

:CALCulate:PARameter[:CATegory]:NOTCh:K

Function Sets or queries the magnification of the notch width analysis function.

Syntax
```
:CALCulate:PARameter[:CATegory]:
NOTCh:K<wsp><NRf>
:CALCulate:PARameter[:CATegory]:
NOTCh:K?
```
<NRf> = Magnification

Example
```
:CALCULATE:PARAMETER:NOTCH:K 2.00
:CALCULATE:PARAMETER:NOTCH:K?
-> +2.00000000E+000
```

Description This is a sequential command.

:CALCulate:PARameter[:CATegory]:NOTCh:TH

Function Sets or queries the threshold value for the notch width analysis function.

Syntax
```
:CALCulate:PARameter[:CATegory]:
NOTCh:TH<wsp><NRf>[DB]
:CALCulate:PARameter[:CATegory]:
NOTCh:TH?
```
<NRf> = Threshold level [dB]

Example
```
:CALCULATE:PARAMETER:NOTCH:
TH 3.00DB
:CALCULATE:PARAMETER:NOTCH:TH?->
+3.00000000E+000
```

Description This is a sequential command.

:CALCulate:PARameter[:CATegory]:NOTCh:TYPE

Function Sets or queries the analysis direction of the notch width analysis function.

Syntax
```
:CALCulate:PARameter[:CATegory]:
NOTCh:TYPE<wsp>PEAK|BOTTom|0|1
:CALCulate:PARameter[:CATegory]:
NOTCh:TYPE?
```
PEAK|0: Performs analysis using the peak level of a waveform as a reference.
BOTTom|1: Performs analysis using the bottom level of a waveform as a reference.
Response 0 = PEAK, 1 = BOTTom

Example
```
:CALCULATE:PARAMETER:NOTCH:
TYPE BOTTOM
:CALCULATE:PARAMETER:NOTCH:TYPE? -> 1
```

Description This is a sequential command.

:CALCulate:PARameter[:CATegory]:POWer:OFFSet

Function Sets or queries the offset value for the POWER analysis function.

Syntax
```
:CALCulate:PARameter[:CATegory]:POWer
:OFFSet<wsp><NRf>[DB]
:CALCulate:PARameter[:CATegory]:POWer
:OFFSet?
```
<NRf> = Offset value [dB]

Example
```
:CALCULATE:PARAMETER:POWER:
OFFSET 1.00DB
:CALCULATE:PARAMETER:POWER:OFFSET?->
+1.00000000E+000
```

:CALCulate:PARameter[:CATegory]:SMSR:BANDwidth|:BWIDth

Function Sets or queries the resolution bandwidth used for side mode power normalization of the SMSR analysis function.

Syntax
```
:CALCulate:PARameter[:CATegory]:SMSR:
BANDwidth|:BWIDth<wsp><nrf>[nm]
:CALCulate:PARameter[:CATegory]:SMSR:
BANDwidth|:BWIDth?
```

Example
```
:CALCULATE:PARAMETER:SMSR:
BANDWIDTH0.1NM
:CALCULATE:PARAMETER:SMSR:BANDWIDTH?
-> +1.00000000E-010
```

Description Setting is possible when the side mode power calculation is set to Normalized.
Use the :CALC:PAR:SMSR:SPOW command to set the side mode power calculation method.

:CALCulate:PARameter[:CATegory]:SMSR:MASK

Function Sets or queries the mask value for the SMSR analysis function.

Syntax
```
:CALCulate:PARameter[:CATegory]:SMSR:
MASK<wsp><NRf>[M]
:CALCulate:PARameter[:CATegory]:SMSR:
MASK?
```
<NRf> = Mask value [m]

Example
```
:CALCULATE:PARAMETER:SMSR:MASK 2.0nm
:CALCULATE:PARAMETER:SMSR:MASK?
->+2.00000000E-009
```

:CALCulate:PARameter[:CATegory]:SMSR:MODE

Function Sets or queries the analysis mode for the SMSR analysis function.

Syntax
```
:CALCulate:PARameter[:CATegory]:SMSR:
MODE<wsp>SMSR1|SMSR2|SMSR3|SMSR4
:CALCulate:PARameter[:CATegory]:SMSR:
MODE?
```

Example
```
:CALCULATE:PARAMETER:SMSR:MODE SMSR1
:CALCULATE:PARAMETER:SMSR:MODE?
->SMSR1
```

:CALCulate:PARameter[:CATegory]:SMSR:MDIFf

Function Sets or queries the value for peak/bottom difference for the SMSR analysis function.

Syntax
```
:CALCulate:PARameter[:CATegory]:SMSR:
MDIFf<wsp><NRf>[DB]
:CALCulate:PARameter[:CATegory]:SMSR:
MDIFf?
```

Example
```
:CALCULATE:PARAMETER:SMSR:
MDIFf 3.00DB
:CALCULATE:PARAMETER:SMSR:MDIFf?
-> +3.00000000E+000
```

Description This is a sequential command.

:CALCulate:PARameter[:CATegory]:SMSR:SMPower

Function Sets or queries the side mode power calculation method of the SMSR analysis function.

Syntax
```
:CALCulate:PARameter[:CATegory]:SMSR:
SMPower<wsp>0|1|TDATa|NORMalized
:CALCulate:PARameter[:CATegory]:SMSR:
SMPower?
```
0|TDATa:
The side mode power is set to the peak level of the side mode.
1|NORMalized:
The side mode power is set to the value obtained by normalizing the peak level of the side mode by the resolution bandwidth.

Example
```
:CALCULATE:PARAMETER:SMSR:
SMPOWER TDAT
:CALCULATE:PARAMETER:SMSR:SMPOWER? ->
0
```

Description Use the :CALC:PAR:SMSR:BAND command to set the resolution bandwidth.

:CALCulate:PARameter[:CATegory]:SWENvelope:K

Function Sets or queries the magnification of the ENVELOPE method–based spectrum width analysis function.

Syntax
```
:CALCulate:PARameter[:CATegory]:
SWENvelope:K<wsp><NRf>
:CALCulate:PARameter[:CATegory]:
SWENvelope:K?
```
<NRf> = Magnification

Example
```
:CALCULATE:PARAMETER:SWENVELOPE:
K 2.00
:CALCULATE:PARAMETER:SWENVELOPE:K? ->
+2.00000000E+000
```

Description This is a sequential command.

:CALCulate:PARameter[:CATegory]:SWENvelope:TH1

Function Sets or queries the search threshold level of the ENVELOPE method–based spectrum width analysis function.

Syntax
```
:CALCulate:PARameter[:CATegory]:SWENv
elope:TH1<wsp><NRf>[DB]
:CALCulate:PARameter[:CATegory]:SWENv
elope:TH1?
```
<NRf> = Search threshold level [dB]

Example
```
:CALCULATE:PARAMETER:SWENVELOPE:
TH1 3.00
:CALCULATE:PARAMETER:SWENVELOPE:
TH1?-> +3.00000000E+000
```

Description This is a sequential command.

:CALCulate:PARameter[:CATegory]:SWENvelope:TH2

Function Sets or queries the threshold level of the ENVELOPE method–based spectrum width analysis function.

Syntax
```
:CALCulate:PARameter[:CATegory]:SWENvelope:TH2<wsp><NRf>[DB]
:CALCulate:PARameter[:CATegory]:SWENvelope:TH2?
```
<NRf> = Threshold level [dB]

Example
```
:CALCULATE:PARAMETER:SWENVELOPE:TH2 10.00db
:CALCULATE:PARAMETER:SWENVELOPE:TH2?-> +1.00000000E+001
```

Description This is a sequential command.

:CALCulate:PARameter[:CATegory]:SWPKrms:K

Function Sets or queries the magnification of the PEAK-RMS method–based spectrum width analysis function.

Syntax
```
:CALCulate:PARameter[:CATegory]:SWPKrms:K<wsp><NRf>
:CALCulate:PARameter[:CATegory]:SWPKrms:K?
```
<NRf> = Magnification

Example
```
:CALCULATE:PARAMETER:SWPKRMS:K 2.00
:CALCULATE:PARAMETER:SWPKRMS:K?-> +2.00000000E+000
```

Description This is a sequential command.

:CALCulate:PARameter[:CATegory]:SWPKrms:TH

Function Sets or queries the threshold level of the PEAK-RMS method–based spectrum width analysis function.

Syntax
```
:CALCulate:PARameter[:CATegory]:SWPKrms:TH<wsp><NRf>[DB]
:CALCulate:PARameter[:CATegory]:SWPKrms:TH?
```
<NRf> = Threshold level [dB]

Example
```
:CALCULATE:PARAMETER:SWPKRMS:TH 3.00db
:CALCULATE:PARAMETER:SWPKRMS:TH?-> +3.00000000E+000
```

Description This is a sequential command.

:CALCulate:PARameter[:CATegory]:SWRMs:K

Function Sets or queries the magnification of the RMS method–based spectrum width analysis function.

Syntax
```
:CALCulate:PARameter[:CATegory]:SWRMS:K<wsp><NRf>
:CALCulate:PARameter[:CATegory]:SWRMS:K?
```
<NRf> = Magnification

Example
```
:CALCULATE:PARAMETER:SWRMS:K2.00
:CALCULATE:PARAMETER:SWRMS;K? -> +2.00000000E+000
```

Description This is a sequential command.

:CALCulate:PARameter[:CATegory]:SWRMs:TH

Function Sets or queries the threshold level of the RMS method–based spectrum width analysis function.

Syntax
```
:CALCulate:PARameter[:CATegory]:SWRMS:TH<wsp><NRf>[DB]
:CALCulate:PARameter[:CATegory]:SWRMS:TH?
```
<NRf> = Threshold level [dB]

Example
```
:CALCULATE:PARAMETER:SWRMS:TH 3.00db
:CALCULATE:PARAMETER:SWRMS:TH?-> +3.00000000E+000
```

Description This is a sequential command.

:CALCulate:PARameter[:CATegory]:SWTHresh:K

Function Sets or queries the magnification of the THRESH method–based spectrum width analysis function.

Syntax
```
:CALCulate:PARameter[:CATegory]:SWTHresh:K<wsp><NRf>
:CALCulate:PARameter[:CATegory]:SWTHresh:K?
```
<NRf> = Magnification

Example
```
:CALCULATE:PARAMETER:SWTHRESH:K 2.00
:CALCULATE:PARAMETER:SWTHRESH:K?-> +2.00000000E+000
```

Description This is a sequential command.

:CALCulate:PARameter[:CATegory]:SWTHresh:MFIT

Function Sets or queries whether to enable the mode fit of the THRESH method-based spectrum width analysis function.

Syntax
```
:CALCulate:PARameter[:CATegory]:SWTHresh:MFIT<wsp>OFF|ON|0|1
:CALCulate:PARameter[:CATegory]:SWTHresh:MFIT?
```
Response 0 = OFF, 1 = ON

Example
```
:CALCULATE:PARAMETER:SWTHRESH:MFIT ON
:CALCULATE:PARAMETER:SWTHRESH:MFIT?->1
```

Description This is a sequential command.

:CALCulate:PARameter[:CATegory]:SWTHresh:TH

Function Sets or queries the threshold level of the THRESH method–based spectrum width analysis function.

Syntax
```
:CALCulate:PARameter[:CATegory]:SWTHresh:TH<wsp><NRf>[DB]
:CALCulate:PARameter[:CATegory]:SWTHresh:TH?
```
<NRf> = Threshold level [dB]

Response ex. Same as above

Example
```
:CALCULATE:PARAMETER:SWTHRESH:TH 3.00DB
:CALCULATE:PARAMETER:SWTHRESH:TH?->+3.00000000E+000
```

Description This is a sequential command.

:CALCulate:PARameter[:CATegory]:WDM:DMASk

Function Sets or queries the channel mask threshold level for the WDM analysis function.

Syntax
```
:CALCulate:PARameter[:CATegory]:WDM:DMASk<wsp><NRf>[DB]
:CALCulate:PARameter[:CATegory]:WDM:DMASk?
```
<NRf> = Threshold level [dB] (–999: Mask OFF)

Example
```
:CALCULATE:PARAMETER:WDM:DMASK -999
:CALCULATE:PARAMETER:WDM:DMASK? ->-9.99000000E+002
```

Description
- Channels the level of which are below this parameter will not be detected as a channel.
- To turn off the channel mask function, set the threshold level to –999.
- This is a sequential command.

:CALCulate:PARameter[:CATegory]:WDM:DTYPe

Function Sets or queries the displayed waveforms of the analysis results for the WDM analysis function.

Syntax
```
:CALCulate:PARameter[:CATegory]:WDM:DTYPe<wsp><display type>
:CALCulate:PARameter[:CATegory]:WDM:DTYPe?
```
<display type>=Type of display

ABSolute = Absolute value display

RELative = Relative value display

MDRift = Drift value display based on the past measurement wavelength

GDRift = Drift value display based on the grid wavelength

Response
- 0 = Absolute value display
- 1 = Relative value display
- 2 = Display drift value using previously measured waveforms as a reference
- 3 = Display drift value using grid wavelength as a reference

Example
```
:CALCULATE:PARAMETER:WDM:DTYPE ABSOLUTE
:CALCULATE:PARAMETER:WDM:DTYPE? -> 0
```

Description This is a sequential command.

:CALCulate:PARameter[:CATegory]:WDM:DUAL

Function Sets or queries the SNR calculation mode for the WDM analysis function.

Syntax
```
:CALCulate:PARameter[:CATegory]:WDM:DUAL<wsp>OFF|ON|0|1
:CALCulate:PARameter[:CATegory]:WDM:DUAL?
```
Response 0 = OFF, 1 = ON

Example
```
:CALCULATE:PARAMETER:WDM:DUAL ON
:CALCULATE:PARAMETER:WDM:DUAL ON? ->1
```

Description
- When this set value is 1 (ON), SNR calculation uses both traces A and B data.
- When this set value is 0 (OFF), SNR calculation uses active trace data.
- This is a sequential command.

:CALCulate:PARameter[:CATegory]:WDM:FALGo

Function Sets or queries the fitting function during level measurement applied to noise level measurements made by the WDM analysis function.

Syntax
```
:CALCulate:PARameter[:CATegory]:WDM:FALGo<wsp><algorithm>
:CALCulate:PARameter[:CATegory]:WDM:FALGo?
```
<algorithm> = Fitting function
LINear = LINEAR
GAUSs = GAUSS
LORenz = LORENZ
3RD = 3RD POLY
4TH = 4TH POLY
5TH = 5TH POLY
Response 0 = LINEAR
1 = GAUSS
2 = LORENZ
3 = 3RD POLY
4 = 4TH POLY
5 = 5TH POLY

Example:
```
CALCULATE:PARAMETER:WDM:FALGO GAUSS
:CALCULATE:PARAMETER:WDM:FALGO? -> 1
```

Description This is a sequential command.

:CALCulate:PARameter[:CATegory]:WDM:IRANge

Function Sets or queries the integral frequency range during signal light power calculation by the WDM analysis function

Syntax
```
:CALCulate:PARameter[:CATegory]:WDM:IRANge<wsp><NRf>
:CALCulate:PARameter[:CATegory]:WDM:IRANge?
```
<NRf> Integral frequency range [GHz]

Example
```
calc:par:wdm:iran 40
calc:par:wdm:iran? -> +4.00000000E+001
```

Description This is a sequential command.

:CALCulate:PARameter[:CATegory]:WDM:IRANge:TH

Function Sets or queries the signal optical power calculation threshold of the WDM analysis function.

Syntax
```
:CALCulate:PARameter[:CATegory]:WDM:IRANge:TH<wsp><NRf>[DB]
:CALCulate:PARameter[:CATegory]:WDM:IRANge:TH?
```
<NRf> = Threshold [dB]

Example
```
:CALCulate:PARameter:WDM:IRANge:TH 3.00db
:CALCulate:PARameter:WDM:IRANge:TH? -> +3.00000000E+000
```

Description This is a sequential command.

:CALCulate:PARameter[:CATegory]:WDM:MAREa

Function Sets or queries the mask range during level measurement applied to noise level measurements made by the WDM analysis function.

Syntax
```
:CALCulate:PARameter[:CATegory]:WDM:MAREa<wsp><NRf>[M]
:CALCulate:PARameter[:CATegory]:WDM:MAREa?
```

Example
```
:CALCULATE:PARAMETER:WDM:MAREA 0.40NM
:CALCULATE:PARAMETER:WDM:MAREA? -> +4.00000000E-010
```

Description This is a sequential command.

:CALCulate:PARameter[:CATegory]:WDM:MDIFf

Function Sets or queries the peak bottom difference of channel detection for the WDM analysis function.

Syntax
```
:CALCulate:PARameter[:CATegory]:WDM:MDIFf<wsp><NRf>[DB]
:CALCulate:PARameter[:CATegory]:WDM:MDIFf?
```
<NRf> = Peak bottom difference [dB]

Example
```
:CALCULATE:PARAMETER:WDM:MDIFF 3.00DB
:CALCULATE:PARAMETER:WDM:MDIFF? -> +3.00000000E+000
```

Description This is a sequential command.

:CALCulate:PARameter[:CATegory]:WDM:MMReset

Function Resets the maximum and minimum of the drift values of the WDM analysis function.

Syntax
```
:CALCulate:PARameter[:CATegory]:WDM:MMReset
```

Example
```
:CALCULATE:PARAMETER:WDM:MMRESET
```

Description
- When "DISPLAY TYPE" (set by the :CALCulate:PARameter[:CATegory]:WDM:DTYPe command is set to other than “DRIFT”, an execution error occurs.
- This is a sequential command.

:CALCulate:PARameter[:CATegory]:WDM:NALGo

Function Sets or queries the measurement algorithm applied to noise level measurements made by the WDM analysis function.

Syntax
```
:CALCulate:PARameter[:CATegory]:WDM:
NALGo<wsp><algorithm>
:CALCulate:PARameter[:CATegory]:WDM:
NALGo?
```
<algorithm> = Algorithm
AFIX|0 = AUTO FIX
MFIX|1 = MANUAL FIX
ACENter|2 = AUTO CENTER
MCENter|3 = MANUAL CENTER
PIT|4 = PIT
Response 0 = AUTO FIX
1 = MANUAL FIX
2 = AUTO CENTER
3 = MANUAL CENTER
4 = PIT

Example
```
:CALCULATE:PARAMETER:WDM:
NALGO ACENTER
:CALCULATE:PARAMETER:WDM:NALGO?-> 2
```

Description This is a sequential command.

:CALCulate:PARameter[:CATegory]:WDM:NARea

Function Sets or queries the measuring range applied to noise level measurements made by the WDM analysis function.

Syntax
```
:CALCulate:PARameter[:CATegory]:WDM:
NARea<wsp><NRf>[M]
:CALCulate:PARameter[:CATegory]:WDM:
NARea?
```
<NRf> = NOISE AREA [m]

Example
```
:CALCULATE:PARAMETER:WDM:
NAREA 0.80NM
:CALCULATE:PARAMETER:WDM:NAREA? ->
+8.00000000E-010
```

Description This is a sequential command.

:CALCulate:PARameter[:CATegory]:WDM:NBW

Function Sets or queries the noise bandwidth for the WDM analysis function.

Syntax
```
:CALCulate:PARameter[:CATegory]:WDM:
NBW<wsp><NRf>[M]
:CALCulate:PARameter[:CATegory]:WDM:
NBW?
```
<NRf> = Noise bandwidth [m]

Example
```
:CALCULATE:PARAMETER:WDM:NBW 0.10NM
:CALCULATE:PARAMETER:WDM:NBW?->
+1.00000000E-010
```

Description This is a sequential command.

:CALCulate:PARameter[:CATegory]:WDM:ODISplay

Function Sets or queries whether the OSNR analysis results of the WDM analysis function are shown.

Syntax
```
:CALCulate:PARameter[:CATegory]:WDM:
ODISplay<wsp>OFF|ON|0|1
:CALCulate:PARameter[:CATegory]:WDM:
ODISplay?
```
OFF|0: Hide
ON|1: Show

Example
```
:CALCulate:PARameter:WDM:ODISplay ON
:CALCulate:PARameter:WDM:ODISplay? ->
1
```

Description This is a sequential command.

:CALCulate:PARameter[:CATegory]:WDM:OSLope

Function Sets or queries whether to enable the function of obtaining the least square approximation line in the WDM analysis function.

Syntax
```
:CALCulate:PARameter[:CATegory]:WDM:
OSLope<wsp>OFF|ON|0|1
:CALCulate:PARameter[:CATegory]:WDM:
OSLope?
```
Response 0 = OFF, 1 = ON

Example
```
:CALCULATE:PARAMETER:WDM:OSLOPE ON
:CALCULATE:PARAMETER:WDM:OSLOPE? -> 1
```

Description
- When this set value is 1 (ON), this instrument calculates the least square approximation line of the peak of each channel and draws it on the waveform screen.
- This is a sequential command.

:CALCulate:PARameter[:CATegory]:WDM:PDISplay

Function Sets or queries whether to display data used for fitting of the WDM analysis function on the waveform screen.

Syntax
```
:CALCulate:PARameter[:CATegory]:WDM:P
DISplay<wsp>OFF|ON|0|1
:CALCulate:PARameter[:CATegory]:WDM:P
DISplay?
```
Response 0 = OFF, 1 = ON

Example
```
:CALCULATE:PARAMETER:WDM:
PDISPLAY ON
:CALCULATE:PARAMETER:WDM:PDISPLAY?
-> 1
```

Description
- When this set value is 1 (ON), data used for fitting is displayed on the waveform screen.
- This is a sequential command.

:CALCulate:PARameter[:CATegory]:WDM:RCH

Function Sets or queries the reference channel used in calculating the offset wavelength/level of the WDM analysis function.

Syntax
```
:CALCulate:PARameter[:CATegory]:WDM:RCH<wsp><integer>
:CALCulate:PARameter[:CATegory]:WDM:RCH?
```
<integer> = Reference channel number (0: channel with the highest level)

Example
```
:CALCULATE:PARAMETER:RCH 10
:CALCULATE:PARAMETER:RCH? -> 10
```

Description
- When this set value is “0,” the channel with the highest level is regarded as the reference channel.
- This is a sequential command.

:CALCulate:PARameter[:CATegory]:WDM:RELation

Function Sets or queries the display format of the wavelength/level relative values for the WDM analysis function.

Syntax
```
:CALCulate:PARameter[:CATegory]:WDM:RELation<wsp>OFFSet|SPACing|0|1
:CALCulate:PARameter[:CATegory]:WDM:RELation?
```
OFFSet|0 = Displays an offset value based on any channel.

SPACing|1 = Displays an offset value relative to a neighboring channel.

Response 0 = OFFSET, 1 = SPACING

Example
```
:CALCULATE:PARAMETER:WDM:RELATION SPACING
:CALCULATE:PARAMETER:WDM:RELATION?
-> 1
```

Description
- When "DISPLAY TYPE" (set by the :CALCulate:PARameter[:CATegory]:WDM:DTYPe command is set to other than “ABSOLUTE”, an execution error occurs.
- This is a sequential command.

:CALCulate:PARameter[:CATegory]:WDM:SPOWer

Function Sets or queries the signal light power calculation method of the WDM analysis function.

Syntax
```
:CALCulate:PARameter[:CATegory]:WDM:SPOWer<wsp>PEAK|INTegral|0|1
:CALCulate:PARameter[:CATegory]:WDM:SPOWer?
```
PEAK|0 = Sets the mode peak to the signal light power

INTegral|1 = Sets the power that integrates the spectrum to the signal light power

Response 0 = PEAK, 1 = INTegral

Example
```
:CALCULATE:PARAMETER:WDM:SPOwer PEAK
:CALCULATE:PARAMETER:WDM:SPOWer?
-> 0
```

Description This is a sequential command.

:CALCulate:PARameter[:CATegory]:WDM:TH

Function Sets or queries the threshold level of channel detection for the WDM analysis function.

Syntax
```
:CALCulate:PARameter[:CATegory]:WDM:TH<wsp><NRf>[DB]
:CALCulate:PARameter[:CATegory]:WDM:TH?
```
<NRf> = Threshold level [dB]

Example
```
:CALCULATE:PARAMETER:WDM:TH 20.00DB
:CALCULATE:PARAMETER:WDM:TH->
+2.00000000E+001
```

Description This is a sequential command.

:CALCulate:PARameter[:CATegory]:WDMSmsr:DMASk

Function Sets or queries the channel mask threshold of the WDM SMSR analysis function.

Syntax
```
:CALCulate:PARameter[:CATegory]:WDMSmsr:DMASk<wsp><NRf>[DB]
:CALCulate:PARameter[:CATegory]:WDMSmsr:DMASk?
```
<NRf> = Threshold [dB](-999: mask OFF)

Example
```
:CALCulate:PARameter:WDMSmsr:DMASk -999
:CALCulate:PARameter:WDMSmsr:DMASk?
-> -9.99000000E+002
```

Description
- Channels whose level is less than or equal to this parameter are not detected as channels.
- To turn off the channel mask function, set the threshold to -999.
- This is a sequential command.

:CALCulate:PARameter[:CATegory]:WDMSmsr:MASK

Function Sets or queries the threshold of the WDM SMSR analysis function.

Syntax
```
:CALCulate:PARameter[:CATegory]:
WDMSmsr:MASK<wsp><NRf>[M]
:CALCulate:PARameter[:CATegory]:
WDMSmsr:MASK?
```
<NRf> = Mask value [m]

Example
```
:CALCulate:PARameter:WDMSmsr:
MASK 2.0nm
:CALCulate:PARameter:WDMSmsr:MASK? ->
+2.00000000E-009
```

Description This is a sequential command.

:CALCulate:PARameter[:CATegory]:WDMSmsr:MDIFf

Function Sets or queries the channel detection peak/bottom difference of the WDM SMSR analysis function.

Syntax
```
:CALCulate:PARameter[:CATegory]:
WDMSmsr:MDIFf<wsp><NRf>[DB]
:CALCulate:PARameter[:CATegory]:
WDMSmsr:MDIFf?
```
<NRf> = Peak/bottom difference [dB]

Example
```
:CALCulate:PARameter:WDMSmsr:
MDIFf 3.00db
:CALCulate:PARameter:WDMSmsr:MDIFf?
-> +3.00000000E+000
```

Description This is a sequential command.

:CALCulate:PARameter[:CATegory]:WDMSmsr:MODE

Function Sets or queries the SMSR analysis mode of the WDM SMSR analysis function.

Syntax
```
:CALCulate:PARameter[:CATegory]:
WDMSmsr:MODE<wsp>SMSR1|SMSR2
:CALCulate:PARameter[:CATegory]:
WDMSmsr:MODE?
```
SMSR1: Analysis mode is set to SMSR1.
SMSR2: Analysis mode is set to SMSR2.

Example
```
:CALCulate:PARameter:WDMSmsr:MODE
SMSR1
:CALCulate:PARameter:WDMSmsr:MODE? ->
SMSR1
```

Description This is a sequential command.

:CALCulate:PARameter[:CATegory]:WDMSmsr:SARea

Function Sets or queries each channel's SMSR analysis range of the WDM SMSR analysis function.

Syntax
```
:CALCulate:PARameter[:CATegory]:
WDMSmsr:SARea<wsp><NRf>[M]
:CALCulate:PARameter[:CATegory]:
WDMSmsr:SARea?
```
<NRf> = Analysis range [m]

Example
```
:CALCulate:PARameter:WDMSmsr:
SARea 2.20nm
:CALCulate:PARameter:WDMSmsr:SARea?
-> +2.20000000E-009
```

Description This is a sequential command.

:CALCulate:PARameter[:CATegory]:WDMSmsr:TH

Function: Sets or queries the channel detection threshold of the WDM SMSR analysis function.

Syntax
```
:CALCulate:PARameter[:CATegory]:WDMSm
sr:TH<wsp><NRf>[DB]
:CALCulate:PARameter[:CATegory]:WDMSm
sr:TH?
```
<NRf> = Threshold [dB]

Example
```
:CALCulate:PARameter:WDMSmsr:
TH 20.00db
:CALCulate:PARameter:WDMSmsr:TH? ->
+2.00000000E+001
```

Description This is a sequential command.

:CALCulate:PARameter[:CATegory]:WFBottom

Function Sets or queries parameters for the WDM FILTER-BTM analysis function.

Syntax
```
:CALCulate:PARameter[:CATegory]:WFBottom<wsp><item>,<paramater>,<data>
:CALCulate:PARameter[:CATegory]:WFBottom?<wsp><item>,<paramater>
```
<item> = Analytical item that sets parameter(s)
<parameter> = Parameter to be set
<data> = Data to be set

<item>	<parameter>	<data>
NWAVelength	ALGO	BOTTom|NPEak|
		NBOTtom|
		GFIT|GRID
	MDIFf	<NRf>[DB]
	TH	<NRf>[DB]
	TBANd	<NRf>[M]
BWAVelength	SW	OFF|ON|0|1
CWAVelength	SW	OFF|ON|0|1
	ALGO	NPEak|NBOTtom
	TH	<NRf>[DB]
SBANd	SW	OFF|ON|0|1
	TH	<NRf>[DB]
EBANd	SW	OFF|ON|0|1
	TH	<NRf>[DB]
	TBANd	<NRf>[M]
RIPPle	SW	OFF|ON|0|1
	TBANd	<NRf>[M]
XTALk	SW	OFF|ON|0|1
	SPACing	<NRf>[M]
	TBANd	<NRf>[M]

Example
```
:CALCULATE:PARAMETER:WFBOTTOM NWAV,ALGO,NPEAK
:CALCULATE:PARAMETER:WFBOTTOM? NWAV,ALGO -> NPE:CALCULATE:PARAMETER:WFBOTTOM BWAVELENGTH,SW,OFF
:CALCULATE:PARAMETER:WFBOTTOM? BWAVELENGTH,SW -> 0
```

Description
- If a non-existing parameter is used for a combination, an execution error occurs (a combination of NWAVelength and SPACing, etc.).
- This is a sequential command.

:CALCulate:PARameter[:CATegory]:WFPeak

Function Sets or queries parameters for the WDM FILTER-PEAK analysis function.

Syntax
```
:CALCulate:PARameter[:CATegory]:WFPeak<wsp><item>,<paramater>,<data>
:CALCulate:PARameter[:CATegory]:WFPeak?<wsp><item>,<paramater>
```
<item> = Analytical item that sets parameter(s)
<parameter> = Parameter to be set
<data> = Data to be set

<item>	<parameter>	<data>
NWAVelength	ALGO	PEAK|MEAN|GFIT|
		GRID
	MDIFf	<NRf>[DB]
	TH	<NRf>[DB]
	TBANd	<NRf>[M]
PWAVelength	SW	OFF|ON|0|1
CWAVelength	SW	OFF|ON|0|1
	TH	<NRf>[DB]
SBANd	SW	OFF|ON|0|1
	TH	<NRf>[DB]
PBANd	SW	OFF|ON|0|1
	TH	<NRf>[DB]
	TBANd	<NRf>[M]
RIPPle	SW	OFF|ON|0|1
	TBANd	<NRf>[M]
XTALk	SW	OFF|ON|0|1
	SPACing	<NRf>[M]
	TBANd	<NRf>[M]

Example
```
:CALCULATE:PARAMETER:WFPEAK NWAV,ALGO,PEAK
:CALCULATE:PARAMETER:WFPEAK? NWAV,ALGO -> PEAK
:CALCULATE:PARAMETER:WFPEAK PWAVELENGTH,SW,OFF
:CALCULATE:PARAMETERWFPEAK? PWAVELENGTH,SW -> 0
```

Description
- If a non-existing parameter is used for a combination, an execution error occurs (a combination of NWAVelength and SPACing, etc.).
- This is a sequential command.

:CALCulate:PARameter:COMMon:MDIFf

Function Sets or queries the peak-bottom difference parameter of channel detection used in the analysis function.

Syntax
```
:CALCulate:PARameter:COMMon:MDIFf<wsp><NRf>[DB]
:CALCulate:PARameter:COMMon:MDIFf?
```

Example
```
:CALCULATE:PARAMETER:COMMON:MDIFF 3.00DB
:CALCULATE:PARAMETER:COMMON:MDIFF-> +3.00000000E+000
```

Description This is a sequential command.

5.4 CALibration Group

:CALibration:ALIGn[:IMMediate]

Function Executes optical axis adjustment of the monochromator using the internal reference light source.

Syntax `:CALibration:ALIGn[:IMMediate]`

Example `:CALIBRATION:ALIGN`

Description This is an overlapable command.

:CALibration:ALIGn:INTernal[:IMMediate]

Function Executes optical axis adjustment of the monochromator using the internal reference light source.

Syntax `:CALibration:ALIGn:INTernal[:IMMediate]`

Example `:CALIBRATION:ALIGN:INTERNAL`

Description This is an overlappable command.

:CALibration:WAVelength:EXTernal[:IMMediate]

Function Performs wavelength calibration using an external reference light source.

Syntax `:CALibration:WAVelength:EXTernal[:IMMediate]`

Example `:CALIBRATION:WAVELENGTH:EXTERNALl`

Description
- The type of the external reference light source to be used for calibration is set using the CALibration:WAVelength:EXTernal:SOURce command.
- The wavelength of the external reference light source to be used for calibration is set using the :CALibration:WAVelength:EXTernal:WAVelength command.
- This is an overlapable command.

:CALibration:WAVelength:EXTernal:SOURce

Function Sets or queries the type of the light source used for external reference light source-based wavelength calibration.

Syntax `:CALibration:WAVelength:EXTernal:SOURce<wsp>LASer|GASCell|0|1|`
`:CALibration:WAVelength:EXTernal:SOURce?`

LASer = An external reference light source is used for the laser

GASCell = A gas cell is used as the external reference light source.

Response 0 = Laser, 1 = Gas cell,

Example `:CALIBRATION:WAVELENGTH:EXTERNALl:SOURCE LASER`
`:CALIBRATION:WAVELENGTH:EXTERNALl:SOURCEe? -> 0`

Description This is a sequential command.

:CALibration:WAVelength:EXTernal:WAVelength

Function Sets or queries the wavelength of the light source used for external reference light source-based wavelength calibration.

Syntax `:CALibration:WAVelength:EXTernal:WAVelength<wsp><NRf>[M]`
`:CALibration:WAVelength:EXTernal:WAVelength?`

<NRf> = Wavelength of the external reference light source [nm]

Example `:CALIBRATION:WAVELENGTH:EXTERNALl:WAVELENGTH 1550.000NM`
`:CALIBRATION:WAVELENGTH:EXTERNALl:WAVELENGTH? -> +1.55000000E-006`

Description This is a sequential command.

:CALibration:WAVelength:INTernal[:IMMediate]

Function Performs wavelength calibration using an internal reference light source.

Syntax
```
:CALibration:WAVelength:INTernal
[:IMMediate]
```

Example
```
:CALIBRATION:WAVELENGTH:INTERNALl
```

Description This is an overlapable command.

:CALibration:ZERO[:AUTO]

Function Sets or queries whether to enable the auto offset function of the level.

Syntax
```
:CALibration:ZERO[:AUTO]<wsp>OFF|ON|0|
1|ONCE
:CALibration:ZERO[:AUTO]?
```
Response 0 = OFF, 1 = ON

Example
```
:CALIBRATION:ZERO ONCE
:CALIBRATION:ZERO? -> 1
```

Description
- If you send this command with the parameter “ONCE” when the sweep is stopped, offset adjustment is performed once. In this case, ON/OFF of this setting does not change.
- The operation of this command is complete at the instant the offset adjustment starts. Therefore, the AQ6374E can execute the next command even while offset adjustment is being performed. You can use :CALibration:ZERO[:AUTO]:STATus? to query the execution status of the offset adjustment.
- This is a sequential command.

:CALibration:ZERO[:AUTO]:INTerval

Function Sets or queries the time interval for executing the Auto Offset function for the level.

Syntax
```
:CALibration:ZERO[:AUTO]:INTerval
<wsp><integer>
:CALibration:ZERO[:AUTO]:INTerval?
```
<integer>= Interval of execution (specified in units of minutes)

Example
```
:CALIBRATION:ZERO:INTERVAL 20
:CALIBRATION:ZERO:INTERVAL? -> 20
```

Description
- When a time is set for this parameter, the auto offset adjustment is performed at the specified time interval starting from the moment of execution.
- This is a sequential command.

:CALibration:ZERO[:AUTO]:STATus?

Function Queries the offset adjustment status.

Syntax
```
:CALibration:ZERO[:AUTO]:STATus?
```
0: The offset adjustment is not being executed.
1: The offset adjustment is being executed.

Example
```
:CALIBRATION:ZERO:STATUS? -> 1
```

Description This is a sequential command.

5.5 DISPlay Group

:DISPlay:COLor

Function Sets or queries the screen color mode.

Syntax `:DISPlay:COLor<wsp><mode>`
`:DISPlay:COLor?`
0 = Black and white mode
1 = Color mode

Example `:DISPLAY:COLOR 1`
`:DISPLAY:COLOR? -> 1`

Description This is a sequential command.

:DISPlay[:WINDow]

Function Sets or queries whether the display is enabled.

Syntax `:DISPlay[:WINDow]<wsp>OFF|ON|0|1`
`:DISPlay[:WINDow]?`
Response 0 = OFF, 1 = ON

Example `:DISPLAY OFF`
`:DISPLAY? -> 0`

Description This is a sequential command.

:DISPlay[:WINDow]:OVIew:POSition

Function Sets or queries the ON/OFF and position of the OVERVIEW display shown during zoom operation.

Syntax `:DISPlay[:WINDow]:OVIew:POSition<wsp>OFF|LEFT|RIGHt|0|1|2`
`:DISPlay[:WINDow]:OVIew:POSition?`
OFF = Display OFF
LEFT = The overview display is on the left of the screen.
RIGHt = The overview display is on the right of the screen.
Response 0 = OFF, 1 = LEFT, 2 = RIGHt

Example `:DISPLAY:OVIEW:POSITION RIGHT`
`:DISPLAY:OVIEW:POSITION? -> 2`

Description This is a sequential command.

:DISPlay[:WINDow]:TEXT:CLEar

Function Clears labels.

Syntax `:DISPlay[:WINDow]:TEXT:CLEar`

Example `:DISPLAY:TEXT:CLEAR`

Description This is a sequential command.

:DISPlay[:WINDow]:TEXT:DATA

Function Sets or queries the labels.

Syntax `:DISPlay[:WINDow]:TEXT:DATA<wsp><string>`
`:DISPlay[:WINDow]:TEXT:DATA?`
<string> = Label character string (56 characters max.)

Example `:DISPLAY:TEXT:DATA "Optical Spectrum Analyzer"`
`:DISPLAY:TEXT:DATA?->`
`Optical Spectrum Analyzer`

Description
- A label character string has a maximum length of 56 characters.If a label of more than 56 characters is specified, characters from and exceeding the 57th will be ignored.
- If there is no label, one space character is returned.
- This is a sequential command.

:DISPlay[:WINDow]:TRACe:X[:SCALe]:CENTer

Function Sets or queries the center wavelength of the X-axis of the display scale.

Syntax `:DISPlay[:WINDow]:TRACe:X[:SCALe]:CENTer<wsp><NRf>[M|HZ]`
`:DISPlay[:WINDow]:TRACe:X[:SCALe]:CENTer?`
<NRf> = Center wavelength [m|Hz]
Response <NRf> [m|Hz|m-1]

Example `:DISPLAY:TRACE:X:CENTER 1550.000NM`
`:DISPLAY:TRACE:X:CENTER?->`
`+1.55000000E-006`

Description
- This is a sequential command.
- To set the position using a wavenumber, enter only the wavenumber (without units) when the X-axis unit is wavenumber.
  The unit is [100×cm^{-1}].
  Example: To specify 1300 cm^{-1}, enter 13.
  If the response is 13, it represents 1300 cm^{-1}.

:DISPlay[:WINDow]:TRACe:X[:SCALe]:INITialize

Function Initializes the X-axis parameters of the display scale.

Syntax `:DISPlay[:WINDow]:TRACe:X[:SCALe]:INITialize`

Example `:DISPLAY:TRACE:X:INITIALIZE`

Description
- The following parameters are initialized based on the measurement scale after this command has been executed.
  ZOOM CENTER, ZOOM SPAN, ZOOM START, ZOOM STOP
- This is a sequential command.

:DISPlay[:WINDow]:TRACe:X[:SCALe]:SMSCale

Function Sets parameters of the current display scale to the measurement scale.

Syntax
```
:DISPlay[:WINDow]:TRACe:X[:SCALe]:
SMSCale
```

Example
```
:DISPLAY:TRACE:X:SMSCALE
```

Description
- The following parameters are initialized based on the display scale after this command has been executed.
  CENTER, SPAN, START, STOP
- This is a sequential command.

:DISPlay[:WINDow]:TRACe:X[:SCALe]:SPAN

Function Sets or queries the span of the X-axis of the display scale.

Syntax
```
:DISPlay[:WINDow]:TRACe:X[:SCALe]:SPAN
<wsp><NRf>[M|HZ]
:DISPlay[:WINDow]:TRACe:X[:SCALe]:SPAN?
```
<NRf> = Span [m|Hz]
Response <NRf> [m|Hz|m^{-1}]

Example
```
:DISPLAY:TRACE:X:SPAN 20.0NM
:DISPLAY:TRACE:X:SPAN? ->
+2.00000000E-008
```

Description
- This is a sequential command.
- To set the position using a wavenumber, enter only the wavenumber (without units) when the X-axis unit is wavenumber.
  The unit is [100×cm^{-1}].
  Example: To specify 1300 cm^{-1}, enter 13.
  If the response is 13, it represents 1300 cm^{-1}.

:DISPlay[:WINDow]:TRACe:X[:SCALe]:SRANge

Function Sets or queries whether to limit an analytical range to the display scale range.

Syntax
```
:DISPlay[:WINDow]:TRACe:X[:SCALe]:
SRANge<wsp>OFF|ON|0|1
:DISPlay[:WINDow]:TRACe:X[:SCALe]:
SRANge?
```
Response 0 = OFF, 1 = ON

Example
```
:DISPLAY:TRACE:X:SRANGE on
:DISPLAY:TRACE:X:SRANGE? -> 1
```

Description This is a sequential command.

:DISPlay[:WINDow]:TRACe:X[:SCALe]:STARt

Function Sets or queries the start wavelength of the X-axis of the display scale.

Syntax
```
:DISPlay[:WINDow]:TRACe:X[:SCALe]:STARt
<wsp><NRf>[M|HZ]
:DISPlay[:WINDow]:TRACe:X[:SCALe]:
STARt?
```
<NRf> = Start wavelength [m|Hz]
Response <NRf> [m|Hz|m^{-1}]

Example
```
:DISPLAY:TRACE:X:START 1540.000NM
:DISPLAY:TRACE:X:START?->
+1.54000000E-006
```

Description
- This is a sequential command.
- To set the position using a wavenumber, enter only the wavenumber (without units) when the X-axis unit is wavenumber.
  The unit is [100×cm^{-1}].
  Example: To specify 1300 cm^{-1}, enter 13.
  If the response is 13, it represents 1300 cm^{-1}.

:DISPlay[:WINDow]:TRACe:X[:SCALe]:STOP

Function Sets or queries the stop wavelength of the X-axis of the display scale.

Syntax
```
:DISPlay[:WINDow]:TRACe:X[:SCALe]:STOP<
wsp><NRf>[M|HZ]
:DISPlay[:WINDow]:TRACe:X[:SCALe]:STOP?
```
<NRf> = Stop wavelength [m|Hz]
Response <NRf> [m|Hz|m^{-1}]

Example
```
:DISPLAY:TRACE:X:STOP 1560.000NM
:DISPLAY:TRACE:X:STOP?->
+1.56000000E-006
```

Description
- This is a sequential command.
- To set the position using a wavenumber, enter only the wavenumber (without units) when the X-axis unit is wavenumber.
  The unit is [100×cm^{-1}].
  Example: To specify 1300 cm^{-1}, enter 13.
  If the response is 13, it represents 1300 cm^{-1}.

:DISPlay[:WINDow]:TRACe:Y:NMASk

Function Sets whether to mask the display of waveforms the level of which is at or below a set threshold level or queries the condition of whether the relevant waveform display is masked.

Syntax
```
:DISPlay[:WINDow]:TRACe:Y:NMASk<wsp>
<NRf>[DB]
:DISPlay[:WINDow]:TRACe:Y:NMASk?
```
<NRf> = Threshold level [dB] (–999: Masking function OFF)

Example
```
:DISPLAY:TRACE:Y:NMASK -999
:DISPLAY:TRACE:Y:NMASK? ->
-9.99000000E+002
```

Description
- The display of waveforms the level of which is at or below this parameter will be masked.
  To turn off the mask function, set the threshold level to –999.
- This is a sequential command.

:DISPlay[:WINDow]:TRACe:Y:NMASk:TYPE

Function Sets or queries the display method when a waveform display at or below a threshold level is masked.

Syntax
```
:DISPlay[:WINDow]:TRACe:Y:NMASk:TYPE
<wsp>VERTical|HORIzontal|0|1
:DISPlay[:WINDow]:TRACe:Y:NMASk:TYPE?
```
VERTical = Waveform display with zero as the mask value or lower
HORizontal = Waveform display with the mask value as the mask value or lower
Response 0 = VERTical, 1 = HORizontal

Example
```
:DISPLAY:TRACE:Y:NMASK:TYPE VERTICAL
:DISPLAY:TRACE:Y:NMASK:TYPE? -> 0
```

Description This is a sequential command.

:DISPlay[:WINDow]:TRACe:Y[:SCALe]:DNUMber

Function Sets or queries the number of display divisions of the level axis.

Syntax
```
:DISPlay[:WINDow]:TRACe:Y[:SCALe]:
DNUMber<wsp>8|10
:DISPlay[:WINDow]:TRACe:Y[:SCALe]:
DNUMber?
```
8, 10 = Number of display divisions

Example
```
:DISPLAY:TRACE:Y:DNUMBER 10
:DISPLAY:TRACE:Y:DNUMBER? -> 10
```

Description This is a sequential command.

:DISPlay[:WINDow]:TRACe:Y1[:SCALe]:BLEVel

Function Sets or queries the base level applied when the main scale of the level axis is linear.

Syntax
```
:DISPlay[:WINDow]:TRACe:Y1[:SCALe]:
BLEVel<wsp><NRf>[W]
:DISPlay[:WINDow]:TRACe:Y1[:SCALe]:
BLEVel?
```
<NRf> = Base level value [W]

Example
```
:DISPLAY:TRACE:Y1:BLEVEL 1.0MW
:DISPLAY:TRACE:Y1:BLEVEL?->
+1.00000000E-003
```

Description
- If a instrument other than W is specified, an execution error occurs.
- This is a sequential command.

:DISPlay[:WINDow]:TRACe:Y1[:SCALe]:PDIVision

Function Sets or queries the main scale of the level axis.

Syntax
```
:DISPlay[:WINDow]:TRACe:Y1[:SCALe]:
PDIVision<wsp><NRf>[DB]
:DISPlay[:WINDow]:TRACe:Y1[:SCALe]:
PDIVision?
```
<NRf> = Level scale [dB]

Example
```
:DISPLAY:TRACE:Y1:PDIV 5.0DB
:DISPLAY:TRACE:Y1:PDIV?->
+5.00000000E+000
```

Description
- If a instrument other than dB is specified, an execution error occurs.
- This is a sequential command.

:DISPlay[:WINDow]:TRACe:Y1[:SCALe]:RLEVel

Function Sets or queries the reference level of the main scale of the level axis.

Syntax
```
:DISPlay[:WINDow]:TRACe:Y1[:SCALe]:
RLEVel<wsp><NRf>[DBM|W]
:DISPlay[:WINDow]:TRACe:Y1[:SCALe]:
RLEVel?
```
<NRf> = Reference level [dB|W]

Example
```
:DISPLAY:TRACE:Y1:RLEVEL -30dbm
:DISPLAY:TRACE:Y1:RLEVEL?->
-3.00000000E+001
```

Description
- When the unit is omitted in the parameter, the reference level is set in dBm if the main scale of the level axis is in the LOG mode or is set in W if it is in the linear mode.
- If the setting condition of the LOG/linear mode of the level axis' main scale does not match the unit specified in the parameter of the command, the parameter of this command is translated matching the LOG/linear mode of the main scale. For example, when the main scale is LOG and you set the reference level to 1m with this command, the reference level is set to 0 dB.
- This is a sequential command.

:DISPlay[:WINDow]:TRACe:Y1[:SCALe]:RPOSition

Function Sets or queries the position of the reference level of the main scale of the level axis.

Syntax
```
:DISPlay[:WINDow]:TRACe:Y1[:SCALe]:
RPOSition<wsp><integer>[DIV]
:DISPlay[:WINDow]:TRACe:Y1[:SCALe]:
RPOSition?
```
<integer> = Position of the reference level

Example
```
:DISPLAY:TRACE:Y1:RPOSITION 10DIV
:DISPLAY:TRACE:Y1:RPOSITION? -> 10
```

Description
- If a value greater than the number of display divisions of the level axis is specified for the position of the reference level, the position of this level is treated as the top of the scale.
- This is a sequential command.

:DISPlay[:WINDow]:TRACe:Y1[:SCALe]:SPACing

Function Sets or queries the scale mode of the main scale of the level axis.

Syntax
```
:DISPlay[:WINDow]:TRACe:Y1[:SCALe]:
SPACing<wsp>LOGarighmic|LINear|0|1
:DISPlay[:WINDow]:TRACe:Y1[:SCALe]:
SPACing?
```
LOGarithmic = LOG scale
LINear = Linear scale
Response 0 = LOGarithmic, 1 = LINear

Example
```
:DISPLAY:TRACE:Y1:SPACING LINIER
:DISPLAY:TRACE:Y1:SPACING? -> 1
```

Description This is a sequential command.

:DISPlay[:WINDow]:TRACe:Y1[:SCALe]:UNIT

Function Sets or queries the units of the main scale of the level axis.

Syntax
```
:DISPlay[:WINDow]:TRACe:Y1[:SCALe]:UNIT
<wsp><unit>
:DISPlay[:WINDow]:TRACe:Y1[:SCALe]:
UNIT?
```
DBM = dBm
W = W
DBM/NM = dBm/nm or dBm/THz
W/NM = W/nm or W/THz
Response 0 = dBm
1 = W
2 = DBM/NM
3 = W/NM

Example
```
:DISPLAY:TRACE:Y1:UNIT DBM/NM
:DISPLAY:TRACE:Y1:UNIT? -> 2
```

Description
- This is a sequential command.
- This command is invalid when wavenumber display is in use (:UNIT:X is 2).

:DISPlay[:WINDow]:TRACe:Y2[:SCALe]:AUTO

Function Sets or queries the automatic setting function of the sub scale of the level axis.

Syntax
```
:DISPlay[:WINDow]:TRACe:Y2[:SCALe]:AUTO
<wsp>OFF|ON|0|1
:DISPlay[:WINDow]:TRACe:Y2[:SCALe]:
AUTO?
```
Response 0 = OFF, 1 = ON

Example
```
:DISPLAY:TRACE:Y2:AUTO ON
:DISPLAY:TRACE:Y2:AUTO? -> 1
```

Description This is a sequential command.

:DISPlay[:WINDow]:TRACe:Y2[:SCALe]:LENGth

Function Sets or queries the parameter of the optical fiber length used when the unit of the subscale of the level axis is dB/km.

Syntax
```
:DISPlay[:WINDow]:TRACe:Y2[:SCALe]:
LENGth<wsp><NRf>[KM]
:DISPlay[:WINDow]:TRACe:Y2[:SCALe]:
LENGth?
```
<NRf> = Length of optical fiber [km]

Example
```
:DISPLAY:TRACE:Y2:LENGTH 99.999KM
:DISPLAY:TRACE:Y2:LENGTH?->
+9.99990000E+001
```

Explanation
- When the unit of the subscale is set to other than “dB/km”, an execution error occurs.
- This is a sequential command.

:DISPlay[:WINDow]:TRACe:Y2[:SCALe]:OLEVel

Function Sets or queries the offset level of the sub scale of the level axis.

Syntax
```
:DISPlay[:WINDow]:TRACe:Y2[:SCALe]:
OLEVel<wsp><NRf>[DB|DB/KM]
:DISPlay[:WINDow]:TRACe:Y2[:SCALe]:
OLEVel?
```
<NRf> = Offset level [dB|dB/km]

Example
```
:DISPLAY:TRACE:Y2:OLEVEL 10DB
:DISPLAY:TRACE:Y2:OLEVEL? ->
+1.00000000E+001
```

Description
- When the unit of the subscale is set to other than “dB” or “dB/km”, an execution error occurs.
- If the unit is not specified in the parameter, dB is set if the subscale of the level axis is in the dB mode or dB/km is set if it is in the dB/km mode.
- If a unit different from the current set unit (:DISPlay[:WINDow]:TRACe:Y2[:SCALe]:UNIT) of the subscale is specified, an execution error occurs.
- This is a sequential command.

:DISPlay[:WINDow]:TRACe:Y2[:SCALe]:PDIVision

Function Sets or queries the sub scale of the level axis.

Syntax
```
:DISPlay[:WINDow]:TRACe:Y2[:SCALe]:
PDIVision<wsp><NRf>[DB|DB/KM|%]
:DISPlay[:WINDow]:TRACe:Y2[:SCALe]:
PDIVision?
```
<NRf> = Level scale [dB|%]

Example
```
:DISPLAY:TRACE:Y2:PDIVISION 5.0%
:DISPLAY:TRACE:Y2:PDIVISION? ->
+5.00000000E+000
```

Description
- If the unit is not specified in the parameter, the set unit of the subscale of the level axis is used as the set unit of this parameter.
- If a unit different from the current set unit (:DISPlay[:WINDow]:TRACe:Y2[:SCALe]:UNIT) of the subscale is specified, an execution error occurs.
- This is a sequential command.

:DISPlay[:WINDow]:TRACe:Y2[:SCALe]:RPOSition

Function Sets or queries the position of the reference level of the sub scale of the level axis.

Syntax
```
:DISPlay[:WINDow]:TRACe:Y2[:SCALe]:
RPOSition<wsp><integer>[DIV]
:DISPlay[:WINDow]:TRACe:Y2[:SCALe]:
RPOSition?
```
<integer> = Position of the reference level

Example
```
:DISPLAY:TRACE:Y2:RPOSITION 10DIV
:DISPLAY:TRACE:Y2:RPOSITION? -> 10
```

Description
- If a value greater than the number of display divisions of the level axis is specified for the position of the reference level, the position of this level is treated as the top of the scale.
- This is a sequential command.

:DISPlay[:WINDow]:TRACe:Y2[:SCALe]:SMINimum

Function Sets or queries the value of the bottom of the scale applied when the subscale of the level axis is set to the linear or % mode.

Syntax
```
:DISPlay[:WINDow]:TRACe:Y2[:SCALe]:
SMINimum<wsp><NRf>[%]
:DISPlay[:WINDow]:TRACe:Y2[:SCALe]:
SMINimum?
```
<NRf> = Value of the bottom of the scale [%]

Example
```
:DISPLAY:TRACE:Y2:SMINIMUM 0%
:DISPLAY:TRACE:Y2:SMINIMUM? ->
+0.00000000E+000
```

Description
- If the unit is not specified in the parameter, the set unit of the subscale of the level axis is used as the set unit of this parameter.
- If a unit different from the current set unit (:DISPlay[:WINDow]:TRACe:Y2[:SCALe]:UNIT) of the subscale is specified, an execution error occurs.
- This is a sequential command.

:DISPlay[:WINDow]:TRACe:Y2[:SCALe]:UNIT

Function Sets or queries the units of the sub scale of the level axis.

Syntax
```
:DISPlay[:WINDow]:TRACe:Y2[:SCALe]:UNIT
<wsp><unit>
:DISPlay[:WINDow]:TRACe:Y2[:SCALe]:
UNIT?
```
<unit> = Units
DB = dB display
LINear = Linear display
DB/KM = dB/km display
% = % display

Response 0 = DB
1 = LINear
2 = DB/KM
3 = %

Example
```
:DISPLAY:TRACE:Y2:UNIT DB
:DISPLAY:TRACE:Y2:UNIT? -> 0
```

Description This is a sequential command.

:DISPlay[:WINDow]:TRACe:Y1[:SCALe]:INITialize

Function Initializes the display zoom condition of the main level scale.

Syntax
```
:DISPlay[:WINDow]:TRACe:Y1[:SCALe]:
INITialize
```

Example
```
:DISPLAY:TRACE:Y1:INITIALIZE
```

Description
- A execution error will occur if the main scale is not displayed.
- This is a sequential command.

:DISPlay[:WINDow]:TRACe:Y2[:SCALe]:INITialize

Function Initializes the display zoom condition of the sub level scale.

Syntax
```
:DISPlay[:WINDow]:TRACe:Y2[:SCALe]:
INITialize
```

Example
```
:DISPLAY:TRACE:Y2:INITIALIZE
```

Description
- A execution error will occur if the sub scale is not displayed.
- This is a sequential command.

5.6 FORMat Group

:FORMat[:DATA]

Function Sets or queries the format used for data transfer.

Syntax
```
:FORMat[:DATA]<wsp>REAL[,64|,32]|ASCii
:FORMat[:DATA]?
```
ASCii = ASCII format (default)
REAL[,64] = REAL format (64bits)
REAL,32 = REAL format (32bits)

Example
```
FORMAT:DATA REAL,64
FORMAT:DATA? -> REAL,64
FORMAT:DATA REAL,32  FORMAT:DATA? ->
REAL,32
FORMAT:DATA ASCII
FORMAT:DATA? -> ASCII
```

Description
- When the format is set to REAL (binary) using this command, the output data of the following commands are produced in the REAL format.
  - :CALCulate:DATA:CGAin?
  - :CALCulate:DATA:CNF?
  - :CALCulate:DATA:CPOWers?
  - :CALCulate:DATA:CSNR?
  - :CALCulate:DATA:CWAVelengths?
  - :TRACe[:DATA]:X?
  - :TRACe[:DATA]:Y?
- The default is ASCII mode.
- When the *RST command is executed, the format is reset to the ASCII mode.
- The ASCII format outputs a list of numerics each of which is delimited by a comma (,). Example: 12345,12345,....
- By default, the REAL format outputs data in fixed length blocks of 64 bits, floating-point binary numerics.
- If “REAL,32” is specified in the parameter, data is output in the 32-bit, floating-point binary form.
- The fixed length block is defined by IEEE 488.2 and consists of “#” (ASCII), one numeric (ASCII) indicating the number of bytes that specifies the length after #, length designation (ASCII), and binary data of a specified length in this order. Binary data consists of a floating-point data string of 8 bytes (64 bits) or 4 bytes (32 bits). Floating-point data consists of lower-order bytes to higher-order bytes.

  E.g.: #18 [eight <byte data>]
  #280[80 <byte data>]
  #48008[8008 <byte data>]
- For data output in the 32-bit floating-point binary form, cancellation of significant digits is more likely to occur in comparison with transfer of data in the 64-bit, floating-point binary form.
- This is a sequential command.

5.7 INITiate Group

:INITiate[:IMMediate]

Function Makes a sweep.

Syntax `:INITiate[:IMMediate]`

Example `:INITIATE`

Description
- You can stop sweep with the :ABORt command.
- The sweep mode (AUTO, SINGLE, or REPEAT) is set using the :INITiate:SMODe command.
- If this command is executed while the sweep mode is in REPEAT (:INITiate:SMODe REPeat), the operation of the command is complete at the instant a sweep starts. In this case, this command is regarded as a sequential command.
- If this command is executed while the sweep mode is one of AUTO and SINGLE, the operation of the command is complete at the instant a sweep ends. In this case, this command is regarded as a command subject to overlapping.

:INITiate:SMODe

Function Sets or queries the sweep mode.

Syntax `:INITiate:SMODe<wsp><sweep mode>`
`:INITiate:SMODe?`
<sweep mode> = Sweep mode
SINGle = SINGLE sweep mode
REPeat = REPEAT sweep mode
AUTO = AUTO sweep mode
Response 1 = SINGle
2 = REPeat
3 = AUTO

Example `:INITIATE:SMODE REPEAT`
`:INITIATE:SMODE? -> 2`

Description This is a sequential command.

5.8 MMEMory Group

Common Items

- To include a directory name in <"filename">, specify the path in the following manner.
- Specification of an absolute path
  When the head of <"file name"> is character “\”, specify the absolute path.
- Relative path specification
  When the head of <"file name"> is any character other than “\”, specify the relative path from the current directory.

The current directory is specified using the :MMEMory:CDIRectory command.

- If INTernal|EXTernal is not specified, access is made to the current drive.
  The current drive is specified using the :MMEMory:CDRive command.
- If a file name extension is omitted when storing a file, an extension corresponding to the data type will be appended to the file name.
- When loading a file, the file name extension can be omitted.

:MMEMory:ANAMe

Function Sets or queries the naming rule for automatic file names.
Syntax `:MMEMory:ANAMe<wsp>NUMBer|DATE|0|1`
`:MMEMory:ANAMe?`
NUMBer|0 Number
DATE|1 Timestamp
Example `:MMEMORY:ANAME DATE`
`:MMEMORY:ANAME? -> 1`
Description This is a sequential command.

:MMEMory:CATalog?

Function Queries a list of all files in the current directory.
Syntax `:MMEMory:CATalog?<wsp>[INTernal|EXTernal][,<directory name>]`
INTernal = Acquires a file list in the current directory of the internal memory.
EXTernal = Acquires a file list in the current directory of the external USB storage.
directory name = Default name
Response <free size>,<file number>,<file name>,<file name>, … ,<file name>
<free size> = <NRf> Disk’s free size [KB] (1KB=1024 bytes))
<file number>= <integer> number of files
<file name> = File name
Example `:MMEMORY:CATALOG? INTERNAL,"\TEST\SAMPLE"`
`-> +1.91176800E+006,2,test0001.wxc,test0002.wxc`
Description This is a sequential command.

:MMEMory:CDIRectory

Function Sets or queries the current directory.
Syntax :MMEMory:CDIRectory<wsp><directory name>
:MMEMory:CDIRectory?
<directory name> = Directory name to be changed
Example :MMEMORY:CDIRECTORY "\test\sample"
:MMEMORY:CDIRECTORY? -> \test\sample
Description This is a sequential command.

:MMEMory:CDRive

Function Sets or queries the current drive.
Syntax `:MMEMory:CDRive<wsp>INTernal|EXTernal`
`:MMEMory:CDRive?`
INTernal = Makes the current drive the internal memory.
EXTernal = Makes the current drive the external USB storage.
Example `:MMEMORY:CDRIVE INTERNAL`
`:MMEMORY:CDRIVE -> INT`
Description This is a sequential command.

:MMEMory:COPY

Function Copies a specified file.
Syntax `:MMEMory:COPY<wsp><"source file name">,INTernal|EXTernal,<"destination file name">[,INTernal| EXTernal]`
<"source file name"> = File name at the copy source
<"destination file name"> = File name at the copy destination
Example `:MMEMORY:COPY "test001.wxc",INTERNAL,"test002.wxc"`
Description This is a sequential command.

:MMEMory:DATA?

Function Queries the data in the specified file.
Syntax `:MMEMory:DATA?<wsp><"file name">[,INTernal| EXTernal]`
<"file name">= Name of the file to be read
Response The data that was read (binary block data of fixed length starting with “#”)
Example `:MMEMORY:DATA? "test.csv",internal -> #18ABCDEFGH`
Description
- Maximum file size that can be sent is 4 MB.
- For the data format of the fixed length blocks, see :FORMat Command.
- This is a sequential command.

:MMEMory:DELete

Function Deletes a specified file.

Syntax `:MMEMory:DELete<wsp><"file name">[,INTernal|EXTernal]`
<"file name"> = Name of a file to be deleted

Example `:MMEMORY:DELETE "test002.wxc",internal`

Description This is a sequential command.

:MMEMory:LOAD:ATRace

Function Loads the specified waveform files (all traces) into traces.

Syntax `:MMEMory:LOAD:ATRace<wsp><”file name”> [,INTernal|EXTernal]`
<"file name"> = Name of file to load
INTernal|EXTernal = Source drive for loading

Example: `MMEMORY:LOAD:ATRACE "test001.csv",internal`

Description This is a sequential command.

:MMEMory:LOAD:DLOGging

Function Loads the specified data logging file.

Syntax `:MMEMory:LOAD:DLOGging<wsp><"filename">[,INTernal|EXTernal]`
<"filename"> = Name of the file to load
INTernal|EXTernal = Source drive to load from

Example `:MMEMORY:LOAD:DLOGGING "test001.LG8",INTERNAL`

Description
- This command is invalid when data logging is in progress.
- This is a sequential command.

:MMEMory:LOAD:PROGram

Function Loads a specified program file into a specified program number.

Syntax `:MMEMory:LOAD:PROGram<wsp><integer>,<"file name">[,INTernal|EXTernal]`
<trace name> = Number of the program into which a file is loaded
<"file name"> = Name of a file to be loaded
INTernal|EXTernal = Drive of source file to be loaded

Example `MMEMORY:LOAD:PROGRAM 1,"test001.pg6",INTERNAL`

Explanation This is a sequential command.

:MMEMory:LOAD:SETTing

Function Loads a specified setting file.

Syntax `:MMEMory:LOAD:SETTing<wsp><"file name">[,INTernal|EXTernal]`
<"file name"> = Name of a file to be loaded
INTernal|LOPpy = Drive of source file to be loaded

Example `MMEMORY:LOAD:SETTING "test001.stc",INTERNAL`

Description This is a sequential command.

:MMEMory:LOAD:TRACe

Function Loads a specified waveform file into a specified trace.

Syntax `:MMEMory:LOAD:TRACe<wsp><trace name>,<"file name">[,INTernal|EXTernal]`
<trace name> = Trace to be loaded
<"file name"> = Name of file to be loaded
INTernal|EXTernal = Drive of source file to load

Example `:MMEMORY:LOAD:TRACE TRA,"test001.wxc",INTERNAL`

Description This is a sequential command.

:MMEMory:MDIRectory

Function Creates a new directory.

Syntax `:MMEMory:MDIRectory<wsp><"directory name">[,INTernal|EXTernal]`
<directory name> = Directory name to be created
INTernal|EXTernal = Destination drive for created directory

Example `:MMEMORY:MDIRECTORY "sample2",INTERNAL`

Description This is a sequential command.

:MMEMory:REMove

Function Readies the USB storage media for removal or queries the readiness status.

Syntax `:MMEMory:REMove`
`:MMEMory:REMove?`
Response 0 = Ready for removal
1 = Not ready

Example `:MMEMORY:REMOVE`
`:MMEMORY:REMOVE -> 1`

:MMEMory:REName

Function Renames a specified file.

Syntax `:MMEMory:REName<wsp><"new file name">,<"old file name">[,INTernal|EXTernal]`
<"new file name">= Name of new file
<"old file name">= Name of old file
INTernal|EXTernal = Target drive

Example `:MMEMORY:RENAME "test001.wxc","test002.wxc",INTERNAL`

Description This is a sequential command.

:MMEMory:STORe:ARESult

Function Stores a variety of analysis results to a specified file.

Syntax `:MMEMory:STORe:ARESult<wsp><"file name">[,INTernal|EXTernal]`
<"file name"> = Name of a file to be saved
INTernal|EXTernal = Save destination drive

Example `:MMEMORY:STORE:ARESULT "test001",INTERNAL`

Description This is a sequential command.

:MMEMory:STORe:ATRace

Function Stores the specified waveform files (all traces) into traces.

Syntax
```
:MMEMory:STORe:ATRace<wsp>
<"file name"> [,INTernal|EXTernal]
```
<"file name"> = Name of file be saved
INTernal|EXTernal = Save destination drive

Example:
```
MMEMORY:STORE:ATRACE "test001.csv",
internal
```

Description This is a sequential command.

:MMEMory:STORe:ATRace:TYPE

Function Sets or queries the file type for saving all trace waveform data.

Syntax
```
:MMEMory:STORe:ATRace:TYPE<wsp>BIN|
CSV|0|1
:MMEMory:STORe:ATRace:TYPE?
```
Response 0=BIN 1=CSV

Example
```
:MMEMory:STORe:ATRace:TYPE CSV
:MMEMory:STORe:ATRace:TYPE? -> 1
```

Description This is a sequential command.

:MMEMory:STORe:DATA

Function Stores a variety of data to a specified file.

Syntax
```
:MMEMory:STORe:DATA<wsp><"file
name">[,INTernal|EXTernal]
```
<"file name"> = Name of a file to be saved
INTernal|EXTernal = Save destination drive

Example
```
:MMEMORY:STORE:DATA "test001",
INTERNAL
```

Description
- The type of data to be stored is specified using the :MMEMory:STORe:DATA:ITEM command.
- Whether to insert data into or overwrite the file with it when storing it is specified using the :MMEMory:STORe:DATA:MODE command.
- This is a sequential command.

:MMEMory:STORe:DATA:ITEM

Function Sets or queries an item to be used when storing data.

Syntax
```
:MMEMory:STORe:DATA:ITEM<wsp>
<item>,OFF|ON|0|1
:MMEMory:STORe:DATA:ITEM?<wsp><item>
```
<item>
DATE = Date/time at the time of storage
LABel = Label
DATA = DATA area data
CONDition = Setting conditions
OWINdow= OUTPUT WINDOW
TRACe = Waveform data
OFF = Do not save
ON = Save
Response 0 = OFF, 1 = ON

Example
```
:MMEMORY:STORE:DATA:ITEM TRACE,OFF
:MMEMORY:STORE:DATA:ITEM? TRACE
-> 0
```

Description This is a sequential command.

:MMEMory:STORe:DATA:MODE

Function Sets whether to insert data into or overwrite an existing file with the data when storing it or queries the condition of whether data is inserted or overwritten.

Syntax
```
:MMEMory:STORe:DATA:MODE<wsp>ADD|
OVER|0|1
:MMEMory:STORe:DATA:MODE?
```
ADD = Insert mode
OVER = Overwrite mode
Response 0 = ADD, 1 = OVER

Example
```
:MMEMORY:STORE:DATA:MODE OVER
:MMEMORY:STORE:DATA:MODE? -> 1
```

Description This is a sequential command.

:MMEMory:STORe:DLOGging

Function Saves the data logging results to a specified file.

Syntax
```
:MMEMory:STORe:DLOGging<wsp>
<"file name">[,INTernal|EXTernal]
```
<"file name"> = Name of the file to save to
INTernal|EXTernal = Drive to save to

Example
```
:MMEMORY:STORE:
DLOGGING "test001",INTERNAL
```

Description
- This command is invalid when data logging is in progress.
- This is a sequential command.

:MMEMory:STORe:DLOGging:CSAVe

Function Sets or queries whether data logging results will be saved to a file in CSV format.

Syntax
```
:MMEMory:STORe:DLOGging:CSAVe<wsp>
OFF|ON|0|1
:MMEMory:STORe:DLOGging:CSAVe?
```
OFF = Data will not be saved to CSV format.
ON = Data will be saved to CSV format.
Response 0 = Off, 1 = On

Example
```
:MMEMORY:STORE:DLOGGING:CSAVE ON
:MMEMORY:STORE:DLOGGING:CSAVE? -> 1
```

Description
- This command is invalid when data logging is in progress.
- This is a sequential command.

:MMEMory:STORe:DLOGging:TSAVe

Function Sets or queries whether temporary saved waveform files will be saved when data logging results is saved.

Syntax
```
:MMEMory:STORe:DLOGging:TSAVe<wsp>
OFF|ON|0|1
:MMEMory:STORe:DLOGging:TSAVe?
```
OFF: Will not be saved
ON: Will be saved
Response 0 = Off, 1 = On

Example
```
:MMEMORY:STORE:DLOGGING:TSAVE ON
:MMEMORY:STORE:DLOGGING:TSAVE? -> 1
```

Description
- This command is invalid when data logging is in progress.
- This is a sequential command.

:MMEMory:STORe:GRAPhics

Function Stores a waveform screen to a specified graphic file.

Syntax `:MMEMory:STORe:GRAPhics<wsp>B&W|COLor|PCOLor,BMP|TIFF,<"file name">[,INTernal| EXTernal]`

B&W|COLor PCOLor = Color mode when saving
- B&W = Black and white mode
- COLor = Color mode
- PCOLor = Preset color (waveforms in color, background in black & white)

BMP|JPG|PNG = Saved format
- BMP = BMP format
- JPG = JPG format
- PNG = PNG format

<"file name"> = Name of a file to be saved

INTernal|EXTernal = Save destination drive

Example `:MMEMORY:STORE:GRAPHICS COLOR,BMP,"test001",INTERNAL`

Description This is a sequential command.

:MMEMory:STORe:PROGram

Function Stores a specified program to a specified file.

Syntax `:MMEMory:STORe:PROGram<wsp><integer>,<"file name">[,INTernal|EXTernal]`

<integer> = Number of a program whose contents are stored

<"file name"> = Name of a file to be saved

NTernal|EXTernal = Save destination drive

Example `:MMEMORY:STORE:PRORAM 1,"test001",INTERNAL`

Explanation This is a sequential command.

:MMEMory:STORe:SETTing

Function Stores setting information to a specified file.

Syntax `:MMEMory:STORe:SETTing<wsp><"file name">[,INTernal|EXTernal]`

<"file name"> = Name of a file to be saved

INTernal|EXTernal = Save destination drive

Example `:MMEMORY:STORE:SETTING "test001",INTERNAL`

Description This is a sequential command.

:MMEMory:STORe:TRACe

Function Stores a specified trace to a specified waveform file.

Syntax `:MMEMory:STORe:TRACe<wsp><trace name>,BIN|CSV,<"file name">[,INTernal|EXTernal]`

<trace name> = Trace to be saved

BIN|CSV = Save format

BIN = Binary format

CSV = Text format)

<"file name"> = Name of file to be saved

INTernal|EXTernal = Save destination drive

Example `:MMEMORY:STORE:TRACE TRA,CSV,"test001",INTERNAL`

Description This is a sequential command.

5.9 SENSe Group

:SENSe:AVERage:COUNt

Function Sets or queries the number of times averaging for each measured point.

Syntax
```
:SENSe:AVERage:COUNt<wsp><integer>
:SENSe:AVERage:COUNt?
```
<integer> = Number of times averaging

Example:
```
:SENSE:AVERAGE:COUNT 100
:SENSE:AVERAGE:COUNT? -> 100
```

Description This is a sequential command.

:SENSe:BANDwidth|:BWIDth[:RESolution]

Function Sets or queries the measurment resolution.

Syntax
```
:SENSe:BANDwidth|:BWIDth[:RESolution]<
wsp><NRf>[M|Hz]
:SENSe:BANDwidth|:BWIDth
[:RESolution]?
```
<NRf> = Measurement resolution [m|Hz]
Response <NRf>[m|Hz|m^{-1}]

Example
```
:SENSE:BANDWIDTH:RESOLUTION 1.5NM
:SENSE:BANDWIDTH? -> +1.50000000E-013
```

Description
- This is a sequential command.
- If the unit is omitted, the value is set using the display unit set on the instrument.
- The resolution is rounded to the closest preset resolution.
  The instrument's preset resolutions are 0.05 nm, 0.1 nm, 0.2 nm, 0.5 nm, 1 nm, 2 nm, 5 nm, and 10 nm.
- To set the position using a wavenumber, enter only the wavenumber (without units) when the X-axis unit is wavenumber.
  The unit is [100×cm^{-1}].
  Example: To specify 1300 cm^{-1}, enter 13.
  If the response is 13, it represents 1300 cm^{-1}.

:SENSe:CHOPper

Function Sets or queries chopper mode.

Syntax
```
:SENSe:CHOPper<wsp>OFF|SWITch|0|2
:SENSe:CHOPper?
```
Response 0 = OFF, 2 = SWITCH

Example
```
:SENSE:CHOPPER SWITCH
:SENSE:CHOPPER? -> 2
```

Explanation
- When the measurement sensitivity setting (:SENSe:SENSe command) is NORMAL HOLD or NORMAL AUTO, Chopper does not function even if chopper mode is turned on with this command.
- This is a sequential command.

:SENSe:CORRection:LEVel:SHIFt

Function Sets or queries the offset value for the level.

Syntax
```
:SENSe:CORRection:LEVel:SHIFt<wsp>
<NRf>[DB]
:SENSe:CORRection:LEVel:SHIFt?
```
<NRf> = Level offset value [dB]

Example
```
:SENSE:CORRECTION:LEVEL:SHIFT 0.2DB
:SENSE:CORRECTION:LEVEL:SHIFT?->
+2.00000000E-001
```

Description This is a sequential command.

:SENSe:CORRection:RVELocity:MEDium

Function Sets or queries whether air or vacuum is used as the wavelength reference.

Syntax
```
:SENSe:CORRection:RVELocity:MEDium
<wsp>AIR|VACuum|0|1
:SENSe:CORRection:RVELocity:MEDium?
```
AIR = Air is assumed to be the reference.
VACuum = Vacuum is assumed to be the reference.
Response 0 = AIR
1 = VACuum

Example
```
:SENSE:CORRECTION:RVELOCITY:
MEDIUM VACUUM
:SENSE:CORRECTION:RVELOCITY:MEDIUM?->
1
```

Description This is a sequential command.

:SENSe:CORRection:WAVelength:SHIFt

Function Sets or queries the offset value for the levelwavelength.

Syntax
```
:SENSe:CORRection:WAVelength:SHIFt
<wsp><NRf>[M]
:SENSe:CORRection:WAVelength:SHIFt?
```
<NRf>= Wavelength offset value [m]

Example
```
:SENSE:CORRECTION:WANELENGTH:
SHIFT 0.05NM
:SENSE:CORRECTION:WANELENGTH:SHIFT?->
+5.00000000E-011
```

Description This is a sequential command.

:SENSe:SENSe:LEVel

Function Sets or queries the sensitivity based on the sensitivity level you want to measure at.

Syntax
```
:SENSe:SENSe:LEVel<wsp><nrf>[DBM]
:SENSe:SENSe:LEVel?
```

Example
```
:SENSE:SENSE:LEVEL -55DBM
:SENSE:SENSE:LEVEL? ->
-5.50000000E-001
```

Description
- By setting the sensitivity level that you want to measure at, the sensitivity closest to that level is automatically selected, and SPEED is also automatically selected.
- This is a sequential command.

:SENSe:SENSe

Function Sets or queries the measurement sensitivity.

Syntax
```
:SENSe:SENSe<wsp><sense>
:SENSe:SENSe?
```
<sense>= Sensitivity setting parameters
NHLD = NORMAL HOLD
NAUT = NORMAL AUTO
NORMal = NORMAL
MID = MID
HIGH1 = HIGH1 or HIGH1
HIGH2 = HIGH2 or HIGH2
HIGH3 = HIGH3 or HIGH3

Response 0 = NHLD
1 = NAUT
2 = MID
3 = HIGH1
4 = HIGH2
5 = HIGH3
6 = NORMAL

Example
```
:SENSE:SENSE MID
:SENSE:SENSE? -> 2
```

Description
- This is a sequential command.
- Sensitivities (x2) indicated above the function menu (N/AUTO(x2)/NORMAL8(x2)/MID(x2)/HIGH1(x2)/HIGH2(x2)/HIGH3(x2)) are set in combination with the :SENS:SWE:SPE command. N/AUTO(x2) corresponds to the following command.
  :SENS:SENS NAUT;
  :SENS:SWE:SPE 2x

:SENSe:SETTing:CORRection

Function Sets or queries the resolution correction function.

Syntax
```
:SENSe:SETTing:CORRection<wsp>OFF|
ON|0|1
:SENSe:SETTing:CORRection?
```
Response 0 = OFF, 1 = ON

Example
```
:SENSE:SETTING:CORRECTION ON
:SENSE:SETTING:CORRECTION? -> 1
```

Explanation This is a sequential command.

:SENSe:SETTing:FIBer

Function Sets or queries the fiber core size mode.

Syntax
```
:SENSe:SETTing:FIBer<wsp>SMALl|
LARGe|0|1
:SENSe:SETTing:FIBer?
```
SMALl= Standard mode
LARGe= Large core size fiber mode
Response 0 = SMALl, 1 = LARGe

Example
```
:SENSE:SETTING:FIBER LARGE
:SENSE:SETTING:FIBER? -> 1
```

Explanation This is a sequential command.

:SENSe:SETTing:SMOothing

Function Sets or queries the Smoothing function.

Syntax
```
:SENSe:SETTing:SMOothing<wsp>OFF|
ON|0|1
:SENSe:SETTing:SMOothing?
```
Response 0 = OFF, 1 = ON

Example
```
:SENSE:SETTING:SMOothing ON
:SENSE:SETTING:SMOothing? -> 1
```

Description This is a sequential command.

:SENSe:SWEep:POINts

Function Sets or queries the number of samples measured.

Syntax
```
:SENSe:SWEep:POINts<wsp><integer>
:SENSe:SWEep:POINts?
```
<integer> = The number of samples to be measured

Example
```
:SENSE:SWEEP:POINTS 20001
:SENSE:SWEEP:POINTS? -> 20001
```

Description
- When the function of automatically setting the sampling number to be measured (:SENSe:SWEep:POINts:AUTO command) is ON, the sampling number to be measured that has been set can be queried.
- When the function of automatically setting the sampling number to be measured (:SENSe:SWEep:POINts:AUTO command) is ON, this command will be automatically set to OFF.
- When the sampling number to be measured is set using this command, the sampling intervals for measurements (:SENSe:SWEep:STEP) will be automatically set.
- This is a sequential command.

:SENSe:SWEep:POINts:AUTO

Function Sets or queries the function of automatically setting the sampling number to be measured.

Syntax
```
:SENSe:SWEep:POINts:AUTO<wsp>OFF|ON|0|1
:SENSe:SWEep:POINts:AUTO?
```
Response 0 = OFF, 1 = ON

Example
```
:SENSE:SWEEP:POINTS:AUTO ON
:SENSE:SWEEP:POINTS:AUTO? -> 1
```

Description
- When the capability to automatically set the sampling number to be measured is set to ON using this command, the sampling number to be measured and the sampling intervals for measurements (:SENSe:SWEep:STEP) will be automatically set.
- This is a sequential command.

:SENSe:SWEep:SPEed

Function Sets or queries the sweep speed.

Syntax
```
:SENSe:SWEep:SPEed<wsp>1x|2x|0|1
:SENSe:SWEep:SPEed?
```
1x|0: Standard
2x|1: Twice as fast as standard
Response 0 = 1x, 1 = 2x

Example
```
:SENSE:SWEEP:SPEED 2x
:SENSE:SWEEP:SPEED? -> 1
```

Description This is a sequential command.

:SENSe:SWEep:STEP

Function Sets or queries the sampling interval for measurements.

Syntax
```
:SENSe:SWEep:STEP<wsp><NRf>[M]
:SENSe:SWEep:STEP?
```
<NRf> = The sampling interval for measurement [m]

Example
```
:SENSE:SWEEP:STEP 1PM
:SENSE:SWEEP:STEP?-> +1.00000000E-012
```

Description
- When the function of automatically setting the sampling interval for measurement (:SENSe:SWEep:POINts:AUTO command) is ON, the sampling number to be measured that has been set can be queried.
- When the function of automatically setting the sampling number to be measured (:SENSe:SWEep:POINts:AUTO command) is ON, this command will be automatically set to OFF.
- When the sampling interval for measurement is set using this command, the sampling intervals for measurements (:SENSe:SWEep:POINts) will be automatically set.
- This is a sequential command.

:SENSe:SWEep:TIME:0NM

Function Sets or queries the time taken from the start to the end of measurements when measurement is made in the 0-nm sweep mode.

Syntax
```
:SENSe:SWEep:TIME:0NM<wsp><integer>[SEC]
:SENSe:SWEep:TIME:0NM?
```
<integer> = Measurement time [sec] (0 = MINIMUM)

Example
```
:SENSE:SWEEP:TIME:0NM 10SEC
:SENSE:SWEEP:TIME:0NM? -> 10
```

Description This is a sequential command.

:SENSe:SWEep:TIME:INTerval

Function Sets or queries the time taken from the start of a sweep to that of the next sweep when repeat sweeps are made.

Syntax
```
:SENSe:SWEep:TIME:INTerval<wsp><integer>[SEC]
:SENSe:SWEep:TIME:INTerval?
```
<integer> = Measurement time [sec] (0 = MINIMUM)

Example
```
:SENSE:SWEEP:TIME:INTERVAL 100sec
:SENSE:SWEEP:TIME:INTERVAL? -> 100
```

Description This is a sequential command.

:SENSe:WAVelength:CENTer

Function Sets or queries the measurement condition center wavelength.

Syntax
```
:SENSe:WAVelength:CENTer<wsp><NRf>[M|HZ]
:SENSe:WAVelength:CENTer?
```
<NRf> = Measurement center wavelength [m]
Response <NRf>[m|Hz|m^{-1}]

Example
```
:SENSE:WAVELENGTH:CENTER 1550.000NM
:SENSE:WAVELENGTH:CENTER?->
+1.55000000E-006
```

Description
- This is a sequential command.
- To set the position using a wavenumber, enter only the wavenumber (without units) when the X-axis unit is wavenumber.
  The unit is [100×cm^{-1}].
  Example: To specify 1300 cm^{-1}, enter 13.
  If the response is 13, it represents 1300 cm-^{1}.

:SENSe:WAVelength:SPAN

Function Sets or queries the measurement condition measurement span.

Syntax
```
:SENSe:WAVelengthSPAN<wsp><NRf>
[M|HZ]
:SENSe:WAVelength:SPAN?
```
<NRf> = Measurement span [m]
Response <NRf>[m|Hz|m^{-1}]

Example
```
:SENSE:WAVELENGTH:SPAN 20.0NM
:SENSE:WAVELENGTH:SPAN?->
+2.00000000E-008
```

Description
- This is a sequential command.
- To set the position using a wavenumber, enter only the wavenumber (without units) when the X-axis unit is wavenumber.
  The unit is [100×cm^{-1}].
  Example: To specify 1300 cm^{-1}, enter 13.
  If the response is 13, it represents 1300 cm^{-1}.

:SENSe:WAVelength:SRANge

Function Sets or queries whether to limit a sweep range to the spacing between line markers L1 and L2.

Syntax
```
:SENSe:WAVelength:SRANge<wsp>OFF|
ON|0|1
:SENSe:WAVelength:SRANge?
```
Response 0 = OFF, 1 = ON

Example
```
:SENSE:WAVELENGTH:SRANGE ON
:SENSE:WAVELENGTH:SRANGE? -> 1
```

Description This is a sequential command.

:SENSe:WAVelength:STARt

Function Sets or queries the measurement condition measurement start wavelength.

Syntax
```
:SENSe:WAVelength:STARt<wsp><NRf>
[M|HZ]
:SENSe:WAVelength:STARt?
```
<NRf>=Measurement center wavelength [m]
Response <NRf> [m|Hz]

Example
```
:SENSE:WAVELENGTH:START 1540.000NM
:SENSE:WAVELENGTH:START?->
+1.54000000E-006
```

Description This is a sequential command.

:SENSe:WAVelength:STOP

Function Sets or queries the measurement condition measurement stop wavelength.

Syntax
```
:SENSe:WAVelengthSTOP<wsp><NRf>
[M|HZ]
:SENSe:WAVelength:STOP?
```
<NRf> = Measurement stop wavelength [m]
Response <NRf> [m|Hz|m^{-1}]

Example
```
:SENSE:WAVELENGTH:STOP 1560.000NM
:SENSE:WAVELENGTH:STOP?->
+1.56000000E-006
```

Description
- This is a sequential command.
- To set the position using a wavenumber, enter only the wavenumber (without units) when the X-axis unit is wavenumber.
  The unit is [100×cm^{-1}].
  Example: To specify 1300 cm^{-1}, enter 13.
  If the response is 13, it represents 1300 cm^{-1}.

5.10 STATus Group

:STATus:OPERation:CONDition?

Function Queries the contents of the operation status condition register.
Syntax :STATus:OPERation:CONDiton?
Example :STATUS:OPERATION:CONDITION? -> 1
Description This is a sequential command.

:STATus:OPERation:ENABle

Function Queries the contents of the operation status Enable register.
Syntax :STATus:OPERation:ENABle<wsp><integer>
:STATus:OPERation:ENABle?
<integer> = Contents of the operation status enable register
Example :STATUS:OPERATION:ENABLE 8
:STATUS:OPERATION:ENABLE? -> 8
Description This is a sequential command.

:STATus:OPERation[:EVENt]?

Function Queries the contents of the operation status Event register.
Syntax :STATus:OPERation[:EVENt]?
Example :STATUS:OPERATION? -> 1
Description This is a sequential command.

:STATus:PRESet

Function Clears the event register and sets all bits of the enable register.
Syntax :STATus:PRESet
Example :STATUS:PRESET
Description
- When this command is executed, the registers will be affected as follows.
- The operation status event register is cleared to “0.”
- All bits of the operation status enable register are set to “0.”
- The questionable status event register is cleared to “0.”
- All bits of the questionable status enable register are set to “0.”
- Even when this command is executed, the standard event status register and standard event status enable register do not change.
- This is a sequential command.

:STATus:QUEStionable:CONDition?

Function Queries the contents of the qestionable status condition register.
Syntax :STATus:QUEStionable:CONDiton?
Example :STATUS:QUESTIONABLE:CONDITION? -> 1
Description This is a sequential command.

:STATus:QUEStionable:ENABle

Function Reads the contents of the questionable status enable register or writes data to this register.
Syntax :STATus:QUEStionable:ENABle<wsp><integer>
:STATus:QUEStionable:ENABle?
<integer> = Contents of the questionable status enable register
Example :STATUS:QUESTIONABLE:ENABLE 8
:STATUS:QUESTIONABLE:ENABLE? -> 8
Description This is a sequential command.

:STATus:QUEStionable[:EVENt]?

Function Reads the contents of the questionable status event register.
Syntax :STATus:QUEStionable[:EVENt]?
Example :STATUS:QUESTIONABLE? -> 1
Description This is a sequential command.

5.11 SYSTem Group

:SYSTem:BUZZer:CLICk

Function Sets or queries whether to sound the buzzer when clicked the key.

Syntax `:SYSTem:BUZZer:CLICk<wsp>OFF|ON|0|1`
`:SYSTem:BUZZer:CLICk?`
Response 0 = OFF, 1 = ON

Example `:SYSTEM:BUZZER:CLICK ON`
`:SYSTEM:BUZZER:CLICK? -> 1`

Description This is a sequential command.

:SYSTem:BUZZer:WARNing

Function Sets or queries whether to sound the buzzer during an alarm.

Syntax `:SYSTem:BUZZer:WARNing<wsp>OFF|ON|0|1`
`:SYSTem:BUZZer:WARNing?`
Response 0 = OFF, 1 = ON

Example `:SYSTEM:BUZZER:WARNING ON`
`:SYSTEM:BUZZER:WARNING? -> 1`

Description This is a sequential command.

:SYSTem:COMMunicate:CFORmat

Function Sets or queries the command format of this unit.

Syntax `:SYSTem:COMMunicate:CFORmat<wsp><mode>`
`:SYSTem:COMMunicate:CFORmat?`
<mode> = Command format
AQ6317 = AQ6317 compatible mode
AQ6374E = AQ6374E mode
Response 0 = AQ6317, 1 = AQ6374E

Example `:SYSTEM:COMMUNICATE:CFORMAT AQ6374E`
`:SYSTEM:COMMUNICATE:CFORMAT? -> 1`

Description
- This command is valid when in AQ6374E mode. This command results in an error when in AQ6317 compatible mode.
- To set the command format while this unit is in the AQ6317-compatible mode, use the following commands.
  Control command
  CFORM* (*: 0 = AQ6317 compatible mode, 1 = AQ6374E mode)
  Query command
  CFORM? (return value: 0 = AQ6317-compatible mode, 1 = AQ6374E mode)
- To use a command to place this unit into the AQ6317-compatible mode, regardless of the status during execution of the command, execute the following command. Note that if this unit has already been in the AQ6317-compatible mode at the time of executing this command, a command error occurs, but you can ignore it.
  :SYSTem:COMMunicate:CFORmat<wsp>AQ6317
- To use a command to place this unit into the AQ6374E mode, regardless of the status during execution of the command, execute the following command. Note that if this unit has already been in the AQ6374E mode at the time of executing this command, a command error occurs, but you can ignore it.
  CFORM1
- This is a sequential command.

:SYSTem:COMMunicate:LOCKout

Function Sets/cancels local lockout.

Syntax
```
:SYSTem:COMMunicate:LOCKout<wsp>
OFF|ON|0|1
:SYSTem:COMMunicate:LOCKout?
```
OFF|0: Cancels local lockout
ON|1: Sets local lockout

Example
```
:SYSTEM:COMMUNICATE:LOCKOUT OFF
:SYSTEM:COMMUNICATE:LOCKOUT? -> 0
```

Description
- This command is valid when the remote interface is the Ethernet interface. An interface message is available for the GP-IB interface.
- During local lockout, if the Ethernet connection is lost, the instrument switches to local mode, regardless of the local lockout status.
- This is a sequential command.

:SYSTem:COMMunicate:RMONitor

Function Sets or queries whether the remote monitor function is enabled.

Syntax
```
:SYSTem:COMMunicate:RMONitor<wsp>
OFF|ON|0|1
:SYSTem:COMMunicate:RMONitor?
```
OFF|0: Disables the remote monitor function
ON|1: Enables the remote monitor function

Example
```
:SYSTEM:COMMUNICATE:RMONITOR OFF
:SYSTEM:COMMUNICATE:RMONITOR? -> 0
```

Description This is a sequential command.

:SYSTem:DATE

Function Sets or queries the system data.

Syntax
```
:SYSTem:DATE<wsp><year>,<month>,<day>
:SYSTem:DATE?
```
<year> = Year
<month> = Month
<day> = Day

Example
```
:SYSTEM:DATE 2023,08,01
:SYSTEM:DATE? -> 2023,08,01
```

Description This is a sequential command.

:SYSTem:DISPlay:UNCal

Function Sets or queries whether to display an alarm message in the event of UNCAL.

Syntax
```
:SYSTem:DISPlay:UNCal<wsp>OFF|ON|0|1
:SYSTem:DISPlay:UNCal?
```
Response 0 = OFF, 1 = ON

Example
```
:SYSTEM:DISPLAY:UNCAL OFF
:SYSTEM:DISPLAY:UNCAL? -> 0
```

Description This is a sequential command.

:SYSTem:ERRor[:NEXT]?

Function Queries data in an error queue and deletes it from the queue.

Syntax
```
:SYSTem:ERRor[:NEXT]?
```
<integer> = Error number

Example
```
:SYSTEM:ERROR? -> 100
```

Description This is a sequential command.

:SYSTem:GRID

Function Sets or queries the instrument's grid setting.

Syntax
```
:SYSTem:GRID<wsp><grid>
:SYSTem:GRID?
```
<grid> = Grid setting
12.5 GHZ = 12.5 GHz Spacing
25 GHZ = 25 GHz Spacing
50 GHZ = 50 GHz Spacing
100 GHZ = 100 GHz Spacing
200 GHZ = 200 GHz Spacing
CUSTom = User setting

Response 0 = 12.5GHz
1 = 25GHz
2 = 50GHz
3 = 100GHz
4 = 200GHz
5 = CUSTom

Example
```
:SYSTEM:GRID 50GHZ
:SYSTEM:GRID? -> 2
```

Description This is a sequential command.
mode.

:SYSTem:GRID:CUSTom:SPACing

Function Sets or queries the grid spacing of the custom grid.

Syntax
```
:SYSTem:GRID:CUSTom:SPACing<wsp><NRf>
[GHZ]
:SYSTem:GRID:CUSTom:SPACing?
```
<NRf> = Grid spacing [GHz]

Example
```
:SYSTem:GRID:CUSTOM:SPACING 12.5
:SYSTem:GRID:CUSTOM:SPACING?->
+1.25000000E+001
```

Description
- When :SYSTem:GRID is CUSTom, an execution error occurs.
- This is a sequential command.

:SYSTem:GRID:CUSTom:STARt

Function Sets or queries the custom grid start wavelength.

Syntax
```
:SYSTem:GRID:CUSTom:STARt<wsp><NRf>
[M|HZ]
:SYSTem:GRID:CUSTom:STARt?
```
<NRf> = Grid start wavelength [m|Hz]

Example
```
:SYSTem:GRID:CUSTOM:START 1550.000NM
:SYSTem:GRID:CUSTOM:START?->
+1.55000000E-006
```

Description
- When :SYSTem:GRID is CUSTom, an execution error occurs.
- This is a sequential command.

:SYSTem:GRID:CUSTom:STOP

Function Sets or queries the custom grid stop wavelength.

Syntax
```
:SYSTem:GRID:CUSTom:STOP<wsp><NRf>
[M|HZ]
:SYSTem:GRID:CUSTom:STOP?
```
<NRf> = Grid stop wavelength [m|Hz]

Example
```
:SYSTEM:GRID:CUSTOM:STOP 1560.000NM
:SYSTEM:GRID:CUSTOM:STOP?->
+1.56000000E-006
```

Description
- When :SYSTem:GRID is something other than CUSTom, an execution error occurs.
- This is a sequential command.

:SYSTem:GRID:REFerence

Function Sets or queries the reference frequency of the instrument's grid setting.

Syntax
```
:SYSTem:GRID:REFerence<wsp><NRf>
[HZ]
:SYSTem:GRID:REFerence?
```
<NRf> = Grid's reference frequency [Hz]

Example
```
:SYSTEM:GRID:REFERENCE 193.1000HZ
:SYSTEM:GRID:REFERENCE?->
+1.93000000E+014
```

Description This is a sequential command.

:SYSTem:INFormation?

Function Queries model-specific information (the model code and special code)

Syntax
```
:SYSTem:INFormation?<wsp><integer>
```
<integer> = The type of instrument specific information to obtain
0 = MODEL code
1 = SPECIAL code

Example
```
:SYSTEM:INFORMATION? 0 ->
AQ6374E-10-L1-FCC-D
:SYSTEM:INFORMATION? 1 -> NONE
```

Description
- Outputs model-specific information (the model code and special code)
- If no SPECIAL mode is present, "NONE" is returned.
- This is a sequential command.

:SYSTem:OLOCk

Function Sets or queries whether keys are locked.

Syntax
```
:SYSTem:OLOCk<wsp>OFF|ON|0|1,
<"password">
:SYSTem:OLOCk?
```
OFF = Not locked (release the lock)
ON = Locked
<"password"> = 4-digit password string
The characters that can be used are numbers from 0 to 9.
Response 0=OFF, 1=ON

Example
```
:SYST:OLOC ON,"1234"
:SYST:OLOC? -> 1
```

Description This is a sequential command.

:SYSTem:PRESet

Function Initializes the unit status.

Syntax
```
:SYSTem:PRESet
```

Example
```
:SYSTEM:PRESET
```

Description This is a sequential command.

:SYSTem:TIME

Function Sets or queries the system time.

Syntax
```
:SYSTem:TIME<wsp><hour>,<minute>,
<second>
:SYSTem:TIME?
```
<hour> = Hour
<minute> = Minute
<second> = Second

Example
```
:SYSTEM:TIME 23,04,01
:SYSTEM:TIME? -> 23,04,1
```

Description This is a sequential command.

:SYSTem:VERSion?

Function Queries the SCPI compatibility version of this unit.

Syntax
```
:SYSTem:VERSion?
```

Example
```
:SYSTEM:VERSION? -> 1999.0
```

Description This is a sequential command.

5.12 TRACe Group

:TRACe:ACTive

Function Sets or queries the active trace.

Syntax
```
:TRACe:ACTive<wsp><trace name>
:TRACe:ACTive?
```
<trace name> = Active trace
(TRA|TRB|TRC|TRD|TRE|TRF|TRG)

Example
```
:TRACE:ACTIVE TRA
:TRACE:ACTIVE? -> TRA
```

Description This is a sequential command.

:TRACe:ATTRibute[:<trace name>]

Function Sets or queries the attributes of the specified trace.

Syntax
```
:TRACe:ATTRibute[:<trace name>]
<wsp><attribute>
:TRACe:ATTRibute[:<trace name>]?
```
<trace name> = trace
(TRA|TRB|TRC|TRD|TRE|TRF|TRG)
<attribute> = Attribute
WRITe = WRITE
FIX = FIX
MAX = MAX HOLD
MIN = MIN HOLD
RAVG = ROLL AVG
CALC = CALC
Response 0 = WRITe
1 = FIX
2 = MAX
3 = MIN
4 = RAVG
5 = CALC

Example
```
:TRACE:ATTRIBUTE:TRA WRITE
:TRACE:ATTRIBUTE:TRA? -> 0
```

Description
- If <trace name> is omitted, the command is executed with respect to the active trace.
- If <trace name> is specified, the specified trace is set as the active trace after the command is exeucted.
- When the attribute is set to a CALC trace, the expression is set using the :CALCulate:MATH command.
- This is a sequential command.

:TRACe:ATTRibute:RAVG[:<trace name>]

Function Sets or queries the number of times for averaging of the specified trace.

Syntax
```
:TRACe:ATTRibute:RAVG[:<trace
name>]<wsp><integer>
:TRACe:ATTRibute:RAVG[:<trace name>]?
```
<trace name> = trace
(TRA|TRB|TRC|TRD|TRE|TRF|TRG)
<integer> = Number of times averaging of ROLL AVG

Example
```
:TRACE:ATTRIBUTE:RAVG:TRA 10
:TRACE:ATTRIBUTE:RAVG:TRA? -> 10
```

Description
- When this command is executed, the attribute of the set trace goes to ROLL AVG.
- If <trace name> is omitted, the command is executed with respect to the active trace.
- If <trace name> is specified, the specified trace is set as the active trace after the command is exeucted.
- This is a sequential command.

:TRACe:COPY

Function Copies the data of a specified trace to another trace.

Syntax
```
:TRACe:COPY<wsp><source trace
name>,<destination trace name>
```
<source trace name> = Copy source trace
<destination trace name>
= Copy trace destination

Example
```
:TRACE:COPY TRA,TRB
```

Description This is a sequential command.

:TRACe[:DATA]:SNUMber?

Function Sets or queries the number of number of data sampled of the specified trace.

Syntax
```
:TRACe[:DATA]:SNUMber?<wsp>
<trace name>
```
<trace name> = Trace from which to acquire data

Example
```
:TRACE:DATA:SNUMBER? -> 50001
```

Description This is a sequential command.

:TRACe[:DATA]:X?

Function Queries the wavelength axis data of the specified trace.

Syntax `:TRACe[:DATA]:X?<wsp><trace name>[,<start point>,<stop point>]`
<trace name>= Trace to be transferred (TRA|TRB|TRC|TRD|TRE|TRF|TRG)
<start point>
= A range of samples to be transferred (starting point) (1 to 200001)
<stop point>
= A range of samples to be transferred (stopping point) (1 to 200001)

Example `:TRACE:X? TRA -> +1.55000000E-006,+1.55001000E-006,+1.55002000E-006,.....`

Description
- The unit is output in wavelength value (m), regardless of whether the X-axis unit is wavelength or frequency.
- If the parameter <start point> or <stop point> is omitted, all sampling data of a specified trace will be output.
  The number of output data can be acquired by executing :TRACe[:DATA]:SNUMber?.
- Data is output in either ASCII or binary form, depending on the setting of :FORMat[:DATA].
- This is a sequential command.

:TRACe[:DATA]:Y?

Function Queries the level axis data of specified trace.

Syntax `:TRACe[:DATA]:Y?<wsp><trace name>[,<start point>,<stop point>]`
<trace name> = Trace to be transferred (TRA|TRB|TRC|TRD|TRE|TRF|TRG)
<start point>
= A range of samples to be transferred (starting point) (1 to 200001)
<stop point>
= A range of samples to be transferred (stopping point) (1 to 200001)
Response
For ASCII data: <NRf>,<NRf>,.....<NRf>
For BINARY data:
'#'<integer><byte num><data byte>

Example `:TRACE:Y? TRA -> -1.00000000E+001, -1.00000000E+001, -1.00000000E+001,....`

Description
- The data is output in order of its wavelength from the shortest level to the longest, regardless of whether the X-axis unit is wavelength or frequency.
- When the level scale is LOG, data is output in LOG values.
- When the level scale is Linear, data is output in linear values.
- If the parameter <start point> or <stop point> is omitted, all sampling data of a specified trace will be output.
  The number of output data can be acquired by executing :TRACe[:DATA]:SNUMber?.
- Data is output in either ASCII or binary form, depending on the setting of :FORMat[:DATA].
- This is a sequential command.

:TRACe[:DATA]:Y:PDENsity?

Function Queries the power spectral density trace data.

Syntax `:TRACe[:DATA]:Y:PDENsity? <wsp>`
`<trace name>,<NRf>[m][,<start point>,`
`<stop point>`
<trace name> = Computation source trace
<NRF> = Normalization bandwidth [m]
<start point>
= Sample range to transfer (start point) (1 to 200001)
<stop point> = Sample range to transfer (stop point) (1 to 200001)

Example `:TRACE:Y:PDEN? TRA,0.1NM`
`-> -5.36017335E+001,-5.36143380E+001,`
`-5.34441639E+001,....`

Description
- When the level scale is set to LOG, LOG values will be output.
- When the level scale is set to linear, linear values will be output.
- If the <start point> and <stop point> parameters are omitted, the entire sample data of the specified trace will be output.
- The data is output in ASCII or BINARY format according to the :FORMat[:DATA] setting.
- This is a sequential command.

:TRACe:DELete

Function Deletes the data of a specified trace.

Syntax `:TRACe:DELete<wsp><trace name>`
<trace name> = Trace to be transferred (TRA|TRB|TRC|TRD|TRE|TRF|TRG)

Example `:TRACE:DELETE TRA`

Description This is a sequential command.

:TRACe:DELete:ALL

Function Clears the data for all traces.

Syntax `:TRACe:DELete:ALL`

Example `:TRACE:DELETE:ALL`

Description This is a sequential command.

:TRACe:STATe[:<trace name>]

Function Sets or queries the display status of the specified trace.

Syntax `:TRACe:STATe[:<trace name>]<wsp>`
`OFF|ON|0|1`
`:TRACe:STATe?`
<trace name> = Trace to be transferred (TRA|TRB|TRC|TRD|TRE|TRF|TRG)
OFF = Hide trace (BLANK)
ON = Makes trace visible (DISP).
Response 0 = OFF, 1 = ON

Example `:TRACE:STATE OFF`
`:TRACE:STATE? -> 0`

Description
- If <trace name> is omitted, the command is executed with respect to the active trace.
- If <trace name> is specified, the specified trace is set as the active trace after the command is exeucted.
- This is a sequential command.

5.13 TRIGger Group

:TRIGger[:SEQuence]:DELay

Function Sets or queries the trigger delay.

Syntax
```
:TRIGger[:SEQuence]:DELay<wsp><NRf>[S]
:TRIGger[:SEQuence]:DELay?
```
<NRf> = delay [sec]

Example
```
:TRIGGER:DELAY 100.0US
:TRIGGER:DELAY? -> +1.00000000E-004
```

Description
- When this command is executed, the external trigger mode becomes enabled. (TRIGger[:SEQuence]:STATe ON)
- This is a sequential command.

:TRIGger[:SEQuence]:GATE:ITIMe

Function Sets or queries sampling interval for gate sampling.

Syntax
```
:TRIGger[:SEQuence]:GATE:ITIMe<wsp><NRf>[S]
:TRIGger[:SEQuence]:GATE:ITIMe?
```
<NRf> = Sampling interval

Response <NRf> = Sampling interval[S]

Example
```
:TRIGGER:SEQUENCE:GATE:ITIME 100ms
:TRIGGER:SEQUENCE:GATE:ITIME? -> +1.00000000E-001
```

Description This is a sequential command.

:TRIGger[:SEQuence]:GATE:LOGic

Function Sets or queries the gate signal logic of gate sampling.

Syntax
```
:TRIGger[:SEQuence]:GATE:LOGic<wsp>POSI|NEGA|0|1
:TRIGger[:SEQuence]:GATE:LOGic?
```
POSI = Sampling is performed when the gate signal is at high level

NEGA = Sampling is performed when the gate signal is at low level

Response 0 = POSI, 1 = NEGA

Example
```
:TRIGGER:SEQUENCE:GATE:LOGIC POSI
:TRIGGER:SEQUENCE:GATE:LOGIC? -> 0
```

Description This is a sequential command.

:TRIGger[:SEQuence]:GATE:MDELay

Function Sets or queries the measure delay of gate sampling.

Syntax
```
:TRIGger[:SEQuence]:GATE:MDELay<wsp><NRf>[S]
:TRIGger[:SEQuence]:GATE:MDELay?
```
<NRf> = Measure delay [sec]

Response <NRf> = Measure delay[S]

Example
```
TRIGGER:SEQUENCE:GATE:MDELay 100ms
:TRIGGER:SEQUENCE:GATE:MDELay?
->+1.00000000E-001
```

Description This is a sequential command.

:TRIGger[:SEQuence]:SLOPe

Function Sets or queries the trigger edge.

Syntax
```
:TRIGger[:SEQuence]:SLOPe<wsp>RISE|FALL|0|1
:TRIGger[:SEQuence]:SLOPe?
```
RISE|0 = RISE

FALL|1 = FALL

Response 0 = RISE, 1 = FALL

Example
```
:TRIGGER:SLOPE RISE
:TRIGGER:SLOPE? -> 0
```

Description
- When this command is executed, the external trigger mode becomes enabled.
- This is a sequential command.

:TRIGger[:SEQuence]:STATe

Function Sets or queries the external trigger mode.

Syntax
```
:TRIGger[:SEQuence]:STATe<wsp>OFF|ON|PHOLd|GATE|0|1|2|3
```
:TRIGger[:SEQuence]:STATe?

OFF|0: External Trigger OFF

ON|1: External trigger mode

PHOLd|2: Peak hold mode

GATE|3: Gate sampling

Response 0 = OFF, 1 = ON, 2 = PHOLd, 3 = GATE

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

Description This is a sequential command.

:TRIGger[:SEQuence]:INPut

Function Sets or queries the signal of the input trigger.

Syntax
```
:TRIGger[:SEQuence]:INPut<wsp>ETRigger|STRigger|SENable|0|1|2
:TRIGger[:SEQuence]:INPut?
```
ETRigger|0: Sampling trigger

STRigger|1: Sweep trigger

SENable|2: Sample enable

Example
```
:TRIGGER:INPUT STRIGGER
:TRIGGER:INPUT? -> 1
```

Description This is a sequential command.

:TRIGger[:SEQuence]:OUTPut

Function Sets or queries the signal of the output trigger.

Syntax
```
:TRIGger[:SEQuence]:OUTPut<wsp>OFF|SSTatus|0|1
:TRIGger[:SEQuence]:OUTPut?
```
OFF|0: OFF
SSTatus|1: Sweep status
Response 0 = OFF, 1 = SSTatus

Example
```
:TRIGGER:OUTPUT SSTATUS
:TRIGGER:OUTPUT? -> 1
```

Description This is a sequential command.

:TRIGger[:SEQuence]:PHOLd:HTIMe

Function Sets or queries the hold time of peak hold mode.

Syntax
```
:TRIGger[:SEQuence]:PHOLd:HTIMe<wsp><NRf>[s]
:TRIGger[:SEQuence]:PHOLd:HTIMe?
```
<NRf> = Hold time [s]

Example
```
:TRIGGER:PHOLD:HTIME 100MS
:TRIGGER:PHOLD:HTIME? ->
+1.00000000E-001
```

Description This is a sequential command.

5.14 UNIT Group

:UNIT:POWer:DIGit

Function Sets or queries the number of decimal places displayed for the level value.

Syntax `:UNIT:POWer:DIGit<wsp>1|2|3`
`:UNIT:POWer:DIGit?`
1, 2, 3 = Number of displayed digits

Example `:UNIT:POWER:DIGIT 3`
`:UNIT:POWER:DIGIT? -> 3`

Description This is a sequential command.

:UNIT:X

Function Sets or queries the units for the X axis.

Syntax `:UNIT:X<wsp>WAVelength|FREQuency|WNUMber|0|1|2`
`:UNIT:X?`
WAVelength|0 = Wavelength
FREQuency|1 = Frequency
WNUMber|2 = Wavenumber
Response 0 = WAVelength
1 = FREQuency
2 = WNUMber

Example `:UNIT:X FREQUENCY`
`:UNIT:X? -> 1`

Description This is a sequential command.

5.15 Common Commands Group

The instrument supports the “Required” common commands listed in the table below.

Cmd	Name	IEEE 488.2 Std.	Commands supported by the AQ6374E
*AAD	Accept Address Command	Option	
*CAL?	Calibration Query	Option	
*CLS	Clear Status Command	Required	Y
*DDT	Define Device Trigger Command	*DT1 option	
*DDT?	Define Device Trigger Query	DT1 option	
*DLF	Disable Listener Function Command	Option	
*DMC	Define Macro Command	Option	
*EMC	Enable Macro Command	Option	
*EMC?	Enable Macro Query	Option	
*ESE	Standard Event Status Enable Command	Required	Y
*ESE?	Standard Event Status Enable Query	Required	Y
*ESR?	Standard Event Status Register Query	Required	Y
*GMC?	Get Macro Contents Query	Option	
*IDN?	Identification Query	Required	Y
*IST?	Individual Status Query	Required for PP1	
*LMC?	Learn Macro Query	Option	
*LRN?	Learn Device Setup Query	Option	
*OPC	Operation Complete Command	Required	Y
*OPC?	Operation Complete Query	Required	Y
*OPT	Option Identification Query	Option	
*PCB	Pass Control Back Command	Required if not C0	
*PMC	Purge Macro Command	Option	
*PRE	Parallel Poll Register Enable Command	Required for PP1	
*PRE?	Parallel Poll Register Enable Query	Required for PP1	
*PSC	Power On Status Clear Command	Option	
*PSC?	Power On Status Clear Query	Option	
*PUD	Protected User Data Command	Option	
*PUD?	Protected User Data Query	Option	
*RCL	Recall Command	Option	
*RDT	Resource DescriptionTransfer Command	Option	
*RDT?	Resource Description Transfer Query	Option	
*RST	Reset Command	Required	Y
*SAV	Save Command	Option	
*SRE	Service Request Enable Command	Required	Y
*SRE?	Service Request Enable Query	Required	Y
*STB?	Read Status Byte Query	Required	Y
*TRG	Trigger Command	Required if DT1	Y
*TST?	Self-Test Query	Required	Y
*WAI	Wait-to-Continue Command	Required	Y

*CLS(Clear Status)

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

Syntax `*CLS`

Example `*CLS`

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

*ESE(Standard Event Status Enable)

Function Sets or queries the standard event enable register.

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

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

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

*ESR?(Standard Event Status Register)

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

Syntax `*ESR?`

Example `*ESR? -> 251`

Description
- The return value of this query is not affected by ESE (Event Status Enable Register).
- This is an overlapping command.

*IDN? (Identification)

Function Queries the instrument type and firmware version.

Syntax `*IDN?`

Example `*IDN? -> YOKOGAWA,AQ6374E,aaaaaaaaa, bb.bb`
aaaaaaaaa: Serial number (9 digit string)
bb.bb: Firmware version

Description
- Outputs 4 field data delimited by a comma.
  Field 1: Manufacturer “YOKOGAWA”
  Field 2: Model “AQ6374E”
  Field 3: Instrument serial number
  Field 4: Firmware version
- This is a sequential command.

*OPC(Operation Complete)

Function Sets or queries bit 0 (OPC) of the standard event status register (ESR) if operations waiting to be processed have all been completed.

Syntax `*OPC`
`*OPC?`

Example `*OPC`
`*OPC? -> 1`

Description
- At the time this command is recognized, the command changes from OCIS (Operation Complete Command Idle State) to OCAS (Operation Complete Command Active State). When the no-operation pending flag is set to “True,” it sets bit 0 (OCR) of ESR and returns to OCIS.
- If any of the following conditions are established, this command is disabled and is forced to return to OCIS.
  Power ON
  Device clear
  *CLS, *RST command
- This is an overlapping command.

*RST (Reset)

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

Syntax `*RST`

Example `*RST`

Description
- Stops operation being processed and returns the instrument to the known set value (default value) immediately.
- This unit’s parameters are cleared.
- The following items will remain the same.
  GP-IB interface status
  GP-IB address
  Output queue
  SRE
  ESE
  Calibration data affecting the instrument’s specifications
- This is an overlapping command.

*SRE(Service Request Enable)

Function Sets or queries the service request enable register.

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

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

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

*STB?(Read Status Byte)

Function Queries the current value of the status byte register.

Syntax
```
*STB?
```

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

Description
- STB will not be cleared even when the contents of the register are read.
- This is an overlapable command.

*TRG(Trigger)

Function Performs a <SINGLE> sweep under the sweep conditions established immediately before receiving the command.

Syntax
```
*TRG
```

Example
```
*TRG
```

Description
- Performs a <SINGLE> sweep regardless of the setting condition of the :INITiate:SMODe command.
- This is an overlapable command.

*TST?(Self Test)

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

Syntax
```
*TST?
```

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

Description
- Of the initialization sequence to be run at startup, this command executes the following operations to output their results. During initialization, the screen maintains the waveform display.
  AMP offset
- Normally returns 0, or 2 for AMP offset error.
- This is a sequential command.

*WAI(Wait to Continue)

Function Prevents the instrument from executing another command until the execution of the current command is complete.

Syntax
```
*WAI
```

Example
```
*WAI
```

Description
- Becomes invalid by device clear.
- Meaningful if subsequent commands are overlapping. Meaningless with other commands.
- This is a sequential command.

5.16 Sample Program

This is an example of remotely controlling this instrument.

Language	: Python
Environment	: Environment in which remote control is possible using the PyVISA library or VISA library
Supported interfaces	: GPIB, ETHERNET (SOCKET), ETHERNET (VXI-11)

The interface used for control can be changed by changing the value of the “iftype” variable in the following program.

Sample Program 1

The program sets the measurement conditions (center wavelength, span, sensitivity, and the sampling number) and then performs a sweep. After completing this sweep, the program executes a thresh-based spectrum width analysis and then outputs the results to the screen.

```
import pyvisa

rm = pyvisa.ResourceManager()

iftype = 0                                          # 0: GPIB, 1:SOCKET, 2:VXI-11

if iftype == 0:
  # === Connect(GP-IB) ===
  board_id = '0'                                    # GP-IB Interface card Address
  osa_adr = '1'                                       # OSA GP-IB Address
  osa = rm.open_resource('GPIB' + board_id + '::' + osa_adr + '::INSTR')
elif iftype == 1:
  # === Connect(Socket) ===
  osa_adr = '192.168.1.100'                         # OSA IP Address
  port = '10001'                                    # OSA remote port num
  osa = rm.open_resource('TCPIP0::' + osa_adr + '::' + port + '::SOCKET')
  # === Set termination ===
  osa.read_termination = '\n'
  osa.write_termination = '\n'
  # === Authentication by OPEN Command ===
  osa.write('open "anonymous"')
  osa.write('')
  out = osa.query("open \"anonymous\"")
  print(out)
  out = osa.query('')
  print(out)
elif iftype == 2:
  # === Connect(VXI-11) ===
  osa_adr = '192.168.1.100'                         # OSA IP Address
  osa = rm.open_resource('TCPIP0::' + osa_adr + '::inst0::INSTR')
```

```
# === Time out setting ===
osa.timeout = 30000                          # Time out = 30sec

# === Set the measurement parameter ===
osa.write('*RST')                            # Setting initialize
osa.write('CFORM1')                          # Command mode set(AQ637x mode)

osa.write(':sens:wav:cent 1550nm')           # sweep center wl
osa.write(':sens:wav:span 10nm')             # sweep span
osa.write(':sens:sens mid')                  # sens mode = MID

osa.write(':sens:sweep:points:auto on')      # Sampling Point = AUTO

# === Sweep execute ===
osa.write(':init:smode 1')                   # single sweep mode
osa.write('*CLS')                            # status clear
osa.write(':init')                           # sweep start

# === Wait for sweep complete ===
while True:
  osa.write(':stat:oper:even?')              # get Operation Event Register
  response = osa.read()
  if int(response.strip()) & 1 == 1:         # Bit0: Sweep status
    break

# === Analysis ===
osa.write(':calc:category swth')           # Spectrum width analysis(THRESH type)
osa.write(':calc')                           # Analysis Execute
osa.write(':calc:data?')                     # get data
response = osa.read()

# === Capture analytical results ===
dblMeanWl = float(response[:16])             # get mean wavelength
dblSpecWd = float(response[17:33])           # get spectrum width

# === Output the result to the screen ===
print("MEAN WL: " + str(dblMeanWl * 1000000000) + " nm")
print("SPEC WD: " + str(dblSpecWd * 1000000000) + " nm")

# === Disconnect ===
osa.close()
```

Sample Program 2

Save an image of the instrument's screen to a PNG file, then use a file transfer command to load the file onto the PC. Save the image on the PC under the file name, "C:\test.png".

```
import pyvisa

rm = pyvisa.ResourceManager()

iftype = 0                                          # 0: GPIB, 1:SOCKET, 2:VXI-11

if iftype == 0:
  # === Connect(GP-IB) ===
  board_id = '0'                                    # GP-IB Interface card Address
  osa_adr = '1'                                     # OSA GP-IB Address
  osa = rm.open_resource('GPIB' + board_id + '::' + osa_adr + '::INSTR')
elif iftype == 1:
  # === Connect(Socket) ===
  osa_adr = '192.168.1.100'                         # OSA IP Address
  port = '10001'                                    # OSA remote port num
  osa = rm.open_resource('TCPIP0::' + osa_adr + '::' + port + '::SOCKET')
  # === Set termination ===
  osa.read_termination = '\n'
  osa.write_termination = '\n'
  # === Authentication by OPEN Command ===
  osa.write('open "anonymous"')
  osa.write('')
  out = osa.query("open \"anonymous\"")
  print(out)
  out = osa.query('')
  print(out)
elif iftype == 2:
  # === Connect(VXI-11) ===
  osa_adr = '192.168.1.100'                         # OSA IP Address
  osa = rm.open_resource('TCPIP0::' + osa_adr + '::inst0::INSTR')

# === Time out setting ===
osa.timeout = 30000                                 # Time out = 30sec

# === send command to OSA ===
osa.write('CFORM1')                                 # Command mode set(AQ637x mode)

# === Save bmp file to internal memory ===
osa.write(':mmem:stor:grap color,png,\"test\",int')

# === Receive binary block data ===
response = osa.query_binary_values(':mmem:data? \"test.png\",int',
datatype='s')
```

```
# === save data to binary file ===
with open('test.png', 'wb') as file:
  file.write(bytes(response))

# === Disconnect ===
osa.close()
```

6.1 Status Registers

This instrument is equipped with the status registers shown in the table below. See the next page for a diagram of all status registers.
This instrument has the following status registers defined by IEEE 488-2 and SCPI:

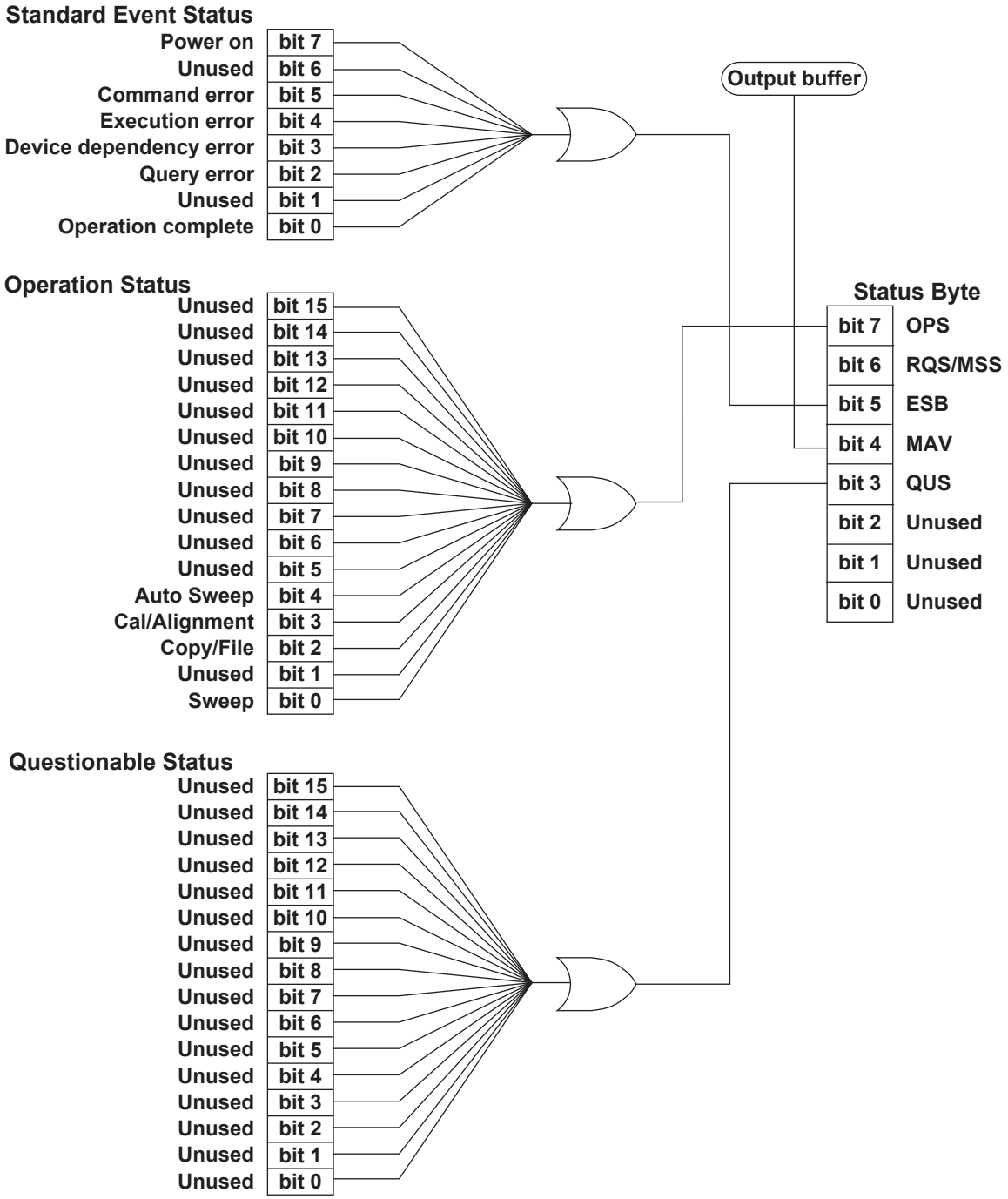
- Status byte registers
- Standard event registers
- Operation status registers
- Questionable status registers

Also, this instrument has an operation status bit (OPS) and a questionable status bit (QUS), each of which contains the summary information of each piece of register information, as the extension bits of the status byte register.

List of Status Registers

Register Name	Description
Status byte registers	Register defined by IEEE 488.2
STB: Status Byte Register	Same as the above
SRE: Service Request Enable Register	Same as the above
Standard event registers	Register defined by IEEE 488.2
ESR: Standard Event Status Register	Same as the above
ESE: Standard Event Status Register	Same as the above
Operation status registers	Provides information on operation execution (such as being swept, or under calibration).
Operation Event Register	A register indicating the presence/absence of an event. Event will be latched.
Operation Event Enable Register	A condition mask register used when the summary bit (OPS) is created.
Questionable status registers	Not assigned yet.
Questionable Event Register	A register indicating the presence/absence of an event. An event will be latched.
Questionable Event Enable Register	A condition mask register used when the summary bit (QUS) is created.

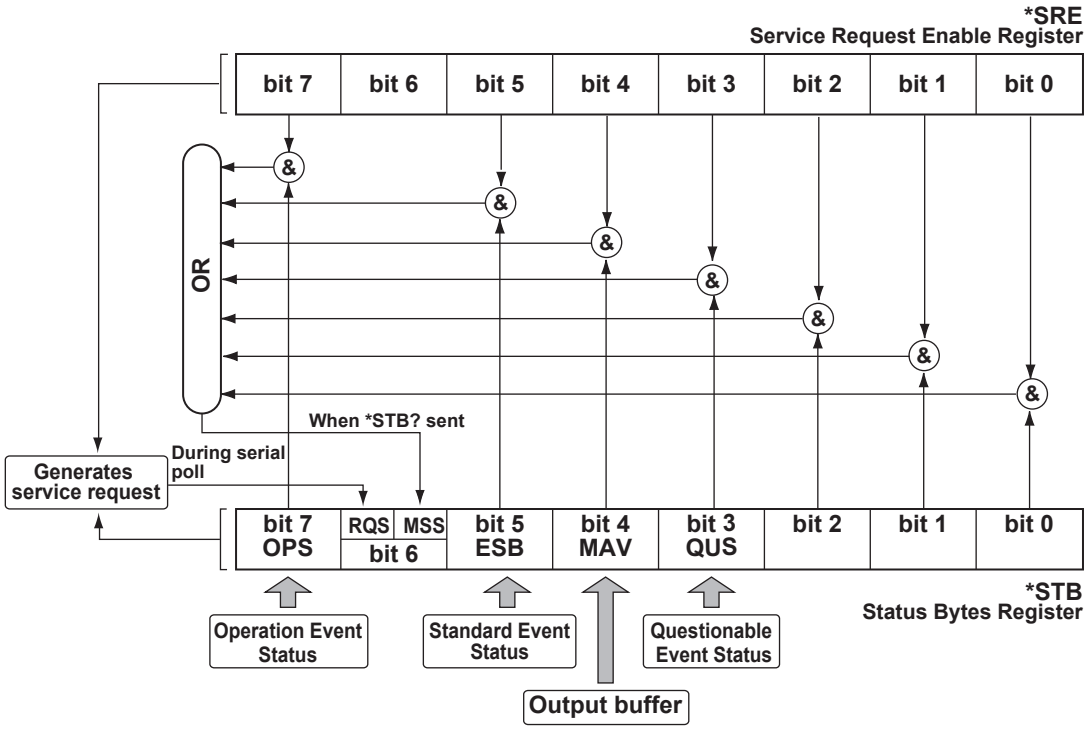
Status Register Overview Diagram



6.2 Status Byte Registers

Structure

The structure of the status byte registers is shown below. The contents and actions of these registers comply with the IEEE 488.2 standards.
Also, the AQ6374E also provides the extended OPS and QUS bits to the status byte register.



Status Byte Register Contents

Bit	Event Name	Description	Decimal Value
Bit 7	OPS	Summary bit of operation status	128
Bit 6	RQS, MSS	"1" if there is more than one service request	64
Bit 5	ESB	Summary bit of standard event status register	32
Bit 4	MAV	"1" if the output buffer contains data	16
Bit 3	QUS	Summary bit of questionable status	8
Bit 2	None	Not used (always 0)	0
Bit 1	None	Not used (always 0)	0
Bit 0	None	Not used (always 0)	0

Status Byte Register

Read

This register can be read by a serial poll or the common *STB? query. Note that the information of bit 6 changes with a different reading method.

- When read by serial polling
  An RQS message is read as bit 6 information.
  After reading, the RQS message will be cleared.
- When read by an *STB? common query
  An MSS summary message is read as bit 6 information.
  Even after reading, the MSS message will be held.

Bits other than bit 6 do not change.
The read action complies with the IEEE 488.2 standard.

Write

The contents of the register will be rewritten only when the status of an assigned status data structure has been changed. The write action complies with the IEEE 488.2 standard.

Clear

All event registers and queues, not including the output queues and MAV bit, will be cleared by the common *CLS command.
The clear action complies with the IEEE 488.2 standard.

Service Request Enable Register

Read

This register can be read by the common *SRE? query.
The value of bit 6, an unassigned bit, is always “0.” The contents of the register are not cleared even when read. The read action complies with the IEEE 488.2 standard.

Write

This register can be written by the common *SRE command.
The set value of bit 6, an unassigned bit, is always ignored. The write action complies with the IEEE 488.2 standard.

Clear

This register will be cleared under any of the following conditions.

- Data “0” is set using the common *SRE command.
- Power ON

The contents of the register are not cleared in the following cases.

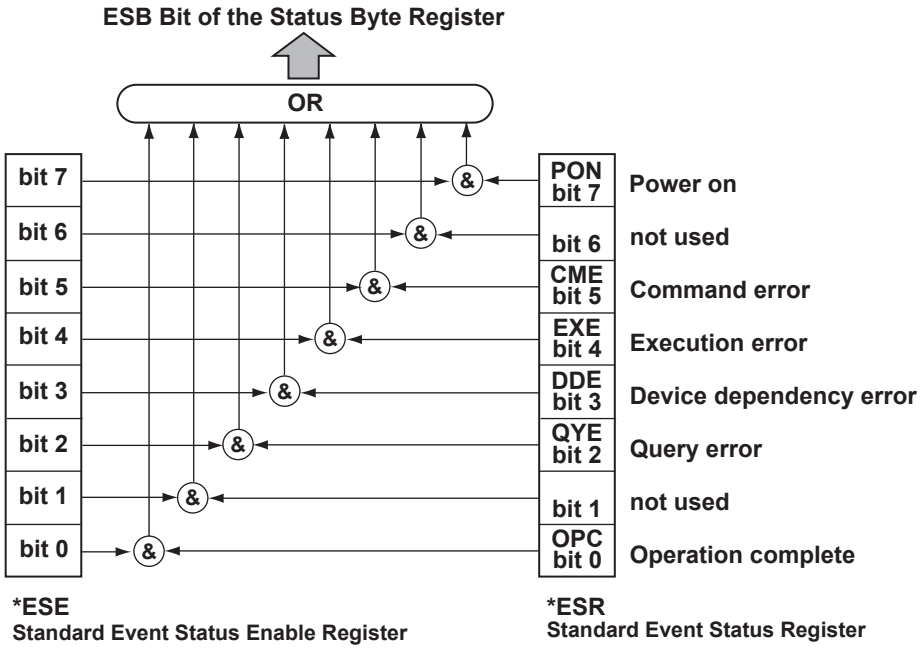
- Receipt of the *RST command
- Receipt of the *CLS command
- Device clear (DCL, SDC)

The clear action complies with the IEEE 488.2 standard.

6.3 Standard Event Status Registers

Structure

The structure of the standard event status registers is shown below. The contents and actions of the registers comply with the IEEE 488.2 standards.



Contents of the Standard Event Status Registers

Bit	Event Name	Description	Decimal Value
Bit 7	PON (Power ON)	Power is turned ON. Set to “1” at startup.	128
Bit 6	None	Not used (always 0)	0
Bit 5	CME (command error)	A syntax error or unrecognizable command is detected. GET is encountered between the 1st byte of a program message and the program message terminator.	32
Bit 4	EXE (Execution error)	Program data following the program header is out of the effective range. Receipt of a program message contradictory to device state.	16
Bit 3	DDE (Device-specific error)	Error caused by an event other than CME, EXE, or QYE.	8
Bit 2	QYE (Query error)	Access to an output queue was made with no output existing. Output queue data was lost.	4
Bit 1	None	Not used (always 0)	0
Bit 0	OPC (operation complete)	Completion of command action: Enabled only when *OPC is received Disabled if *OPC? is received	1

Standard Event Status Register

Read

This register can be read by the common *ESR? query.

Its contents will be cleared when read. The read action complies with the IEEE 488.2 standard.

Write

Contents of the register can be cleared. The register can be cleared but not written to.

Clear

This register will be cleared under any of the following conditions.

- Common *CLS command
- Common *ESR? query

The clear action complies with the IEEE 488.2 standard.

Standard Event Status Enable Register

Read

This register can be read by the common *ESE? query.

The read action complies with the IEEE 488.2 standard.

Write

This register can be written by the common *ESE command.

The write action complies with the IEEE 488.2 standard.

Clear

This register will be cleared under any of the following conditions.

- Data “0” is set using the common *ESE command.
- Power ON

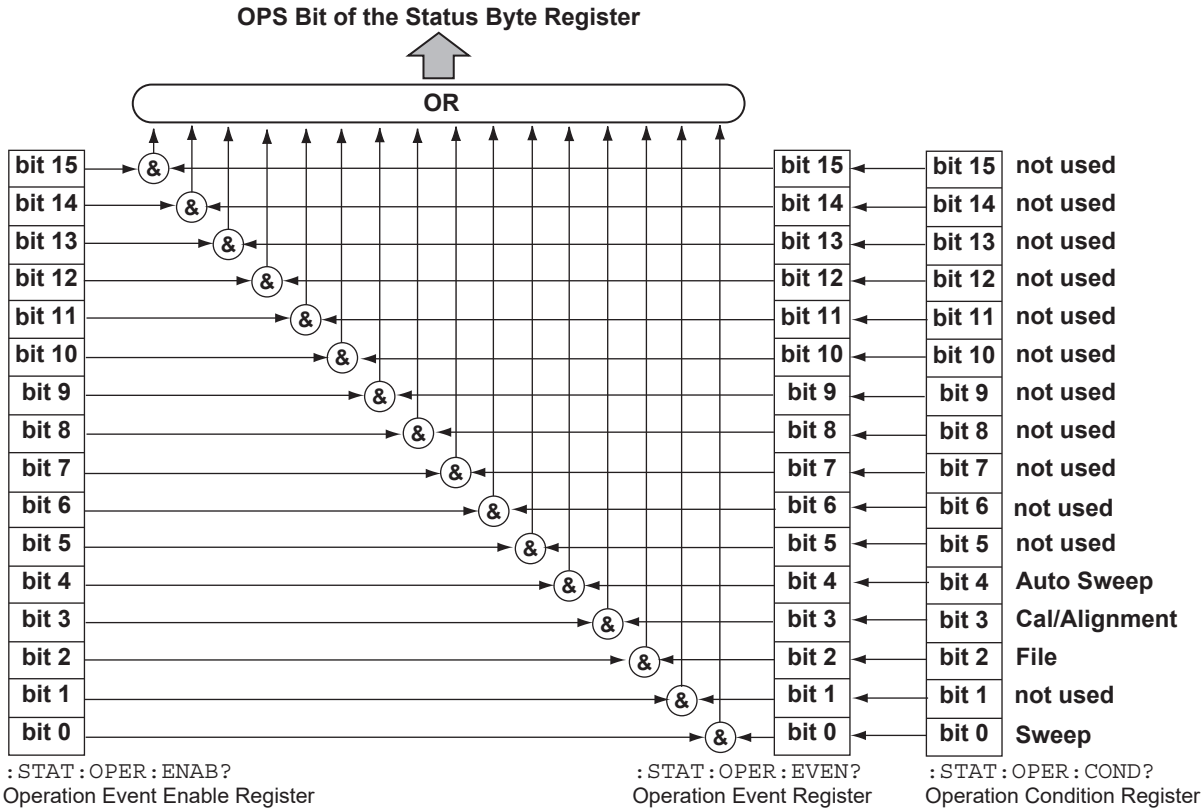
The register cannot be cleared in the following cases.

- Receipt of the *RST command
- Receipt of the *CLS command
- Device clear (DCL, SDC)

The clear action complies with the IEEE 488.2 standard.

6.4 Operation Status Registers

Operation status registers report the operation status of the instrument. The operation condition registers indicate the instrument's condition. A change in an operation condition register is latched into the operation event register. The user can refer to the operation event register to view changes in the operation status. The summary information of the instrument event register is set to the OPS bit of the status byte register. In this case, only statuses corresponding to bits specified as "1" in the operation enable register are included in the summary information.



Contents of the Operation Status Register

Bit	Event Name	Description	Decimal Value
Bit 15	Not used	Spare (always 0)	0
Bit 14	Not used	Spare (always 0)	0
Bit 13	Not used	Spare (always 0)	0
Bit 12	Not used	Spare (always 0)	0
Bit 11	Not used	Spare (always 0)	0
Bit 10	Not used	Spare (always 0)	0
Bit 9	Not used	Spare (always 0)	0
Bit 8	Not used	Spare (always 0)	0
Bit 7	Not used	Spare (always 0)	0
Bit 6	Not used	Spare (always 0)	0
Bit 5	Not used	Spare (always 0)	0
Bit 4	Auto Sweep	Completion of auto sweep running action	16
Bit 3	Cal/Alignment	Completion of wavelength calibration, alignment or resolution calibration	8
Bit 2	File	Completion of file operation	4
Bit 1	Not used	Spare (always 0)	0
Bit 0	Sweep	Completion of a sweep	1

Operation Condition Register

Read

This register can be read by the :STATus:OPERation:CONDition? query command.
Its contents will not be cleared even when read.

Write

The register sets or resets a bit corresponding to a change in the status of the instrument only when that change occurs. It cannot be written to.

Clear

The register cannot be cleared.

Operation Event Register

Read

This register can be read by the :STATus:OPERation[:EVENt?] query command.
Its contents will be cleared when read.

Write

Contents of the register can be cleared. The register can be cleared but not written to.
<Clear>
This register will be cleared under any of the following conditions.

- A read using the :STATus:OPERation[:EVENt?] query command
- An initialization by the :STATus:PRESet command
- The *CLS common command
- Power ON

Operation event enable register

Read

This register can be read by the :STATus:OPERation:ENABle? query command.

Write

The register can be written by the :STATus:OPERation:ENABle command.

Clear

This register will be cleared under any of the following conditions.

- When the instrument is initialized with the :STATus:PRESet command
- Power ON

The register cannot be cleared in the following cases.

- Receipt of the *RST command
- Receipt of the *CLS command
- Device clear (DCL, SDC)

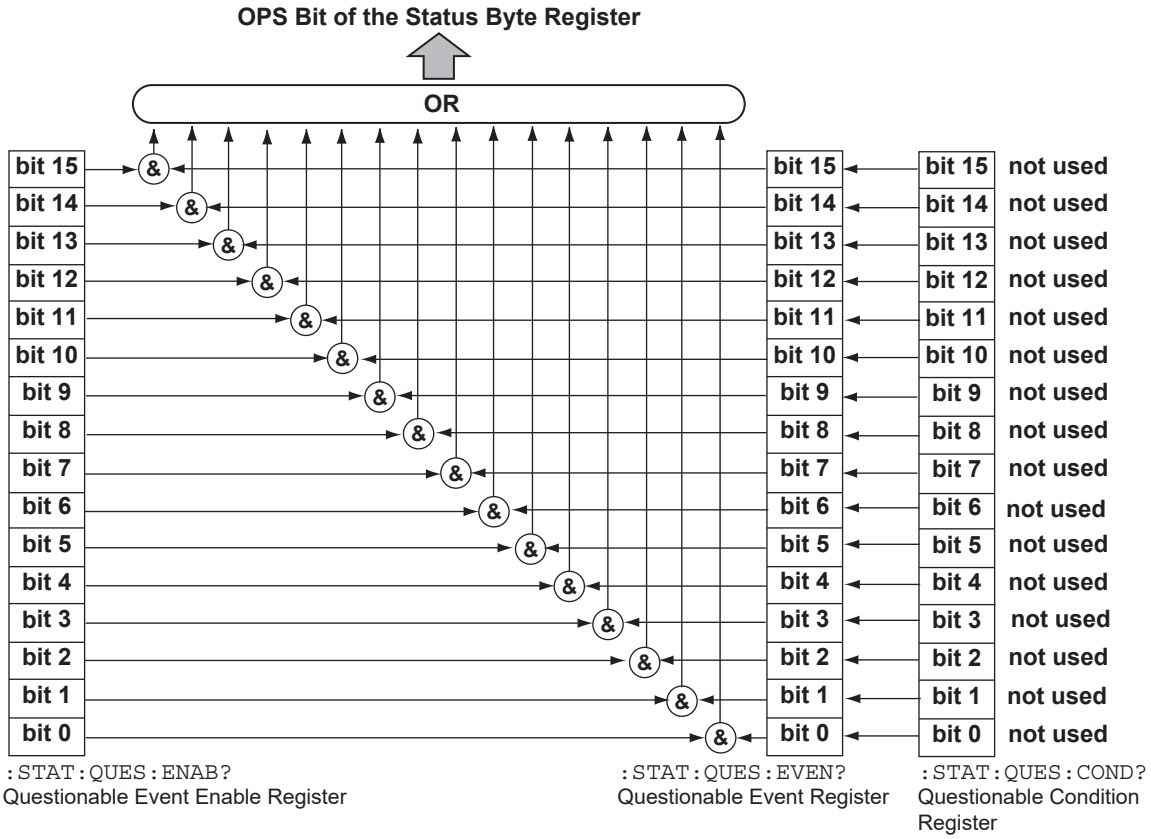
6.5 Questionable Status Registers

The questionable status registers report the questionable status of the instrument. All bits of these registers are unassigned. However, the register read/write operations are performed normally. The summary information of an event register will be set to the QUS bit of the status byte register.

Structure

The structure of the questionable status registers is shown below.

Structure of the Questionable Status Registers



Contents of the Questionable Status Registers

Bit	Event Name	Description	Decimal Value
Bit 0–15	Not used	Spare (always 0)	0

Questionable Condition Register

Read

This register can be read by the :STATus:QUEStionable:CONDition? query command. Its contents will not be cleared even when read.

Write

The register sets or resets a bit corresponding to a change in the status of the instrument only when that change occurs. It cannot be written to.

Clear

The register cannot be cleared.

Questionable Event Register

Read

This register can be read by the :STATus:QUEStionable[:EVENt?] query command.
Its contents will be cleared when read.

Write

Contents of the register can be cleared. The register can be cleared but not written to.

Clear

This register will be cleared under any of the following conditions.

- A read using the :STATus:QUEStionable[:EVENt?] query command
- Initialization by the :STATus:PRESet command
- Common *CLS command
- Power ON

Questionable Event Enable Register

Read

This register can be read by the :STATus:QUEStionable:ENABle? query command.

Write

The register can be written to by the :STATus:QUEStionable:ENABle command.

Clear

This register will be cleared under any of the following conditions.

- When the instrument is initialized with the :STATus:PRESet command.
- Power ON

The register cannot be cleared in the following cases.

- Receipt of the *RST command
- Receipt of the *CLS command
- Device clear (DCL, SDC)

7.1 Editing a Program

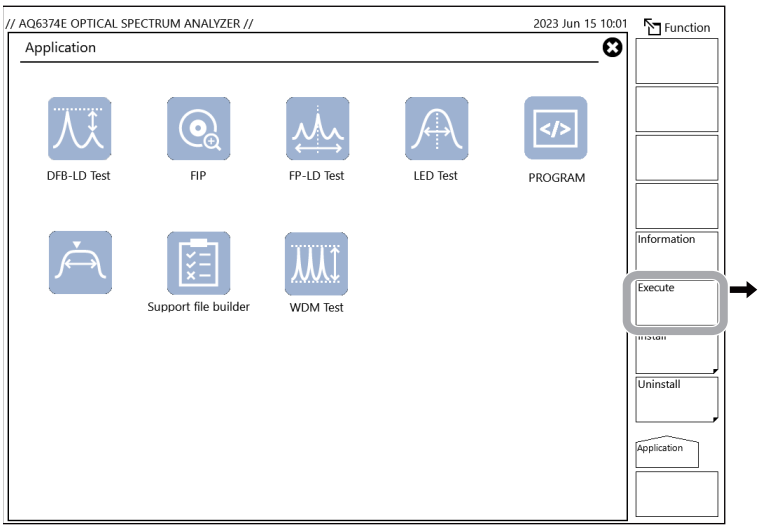
To use the program functions, a program must be pre-registered in the instrument.

Procedure

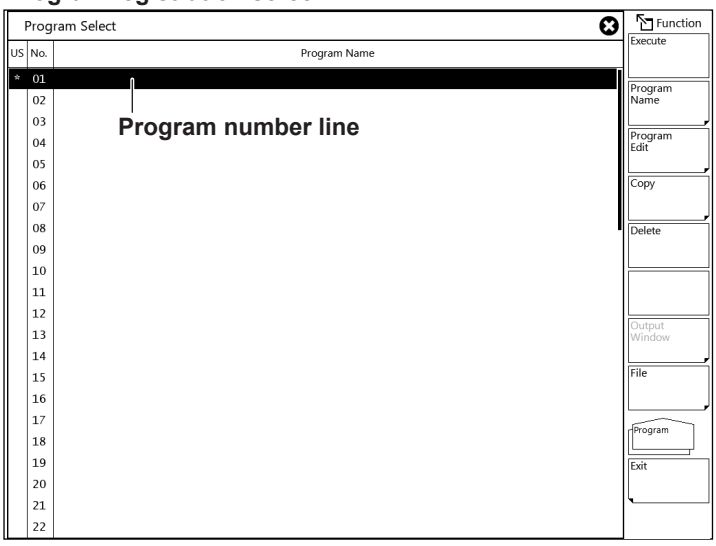
Registering a Program

1. Press **APP**. The APP menu appears.
2. Double-tap **Program**, or tap the Program icon and then tap **Execute**. A program registration screen appears.
3. On the registration screen, tap a program number line.

APP menu

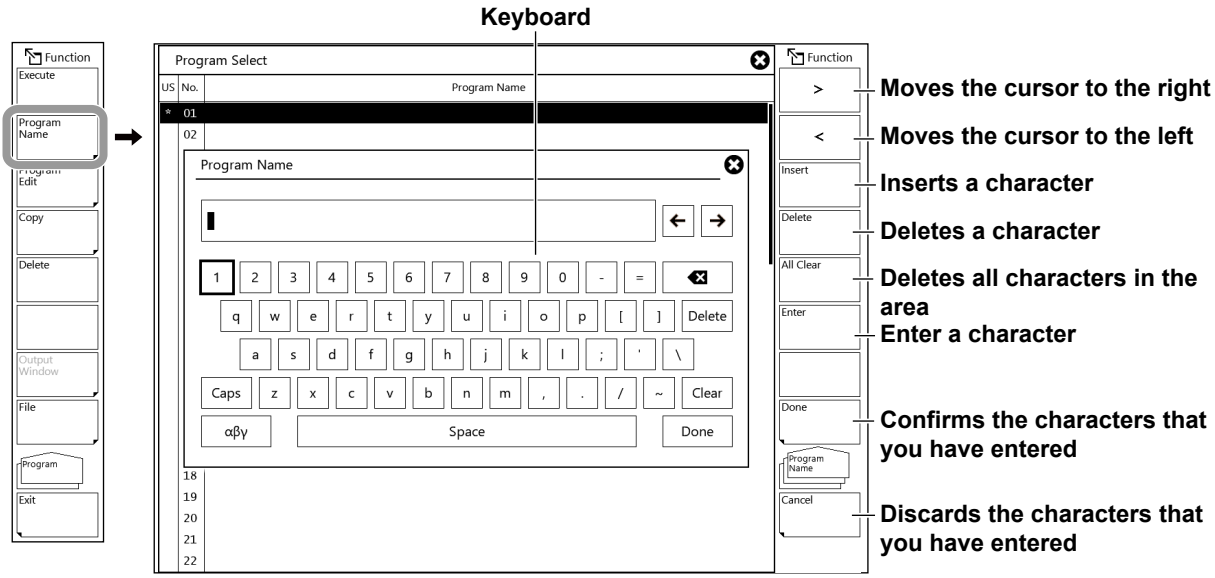


Program registration screen



Assigning a Program Name

4. Tap **Program Name**. A program name input screen appears.
5. Tap a position in the character input area where you want to input characters to move the cursor to that position. You can also use the function menu to move the cursor, insert characters, delete all characters, and so on.
6. Tap a character on the keyboard to input the character. Tap Enter on the function menu to enter the character selected on the keyboard.
7. After entering the program name, tap **Done**. The program name is confirmed, and the screen returns to the program registration screen. The program name you assigned appears on the program registration screen.



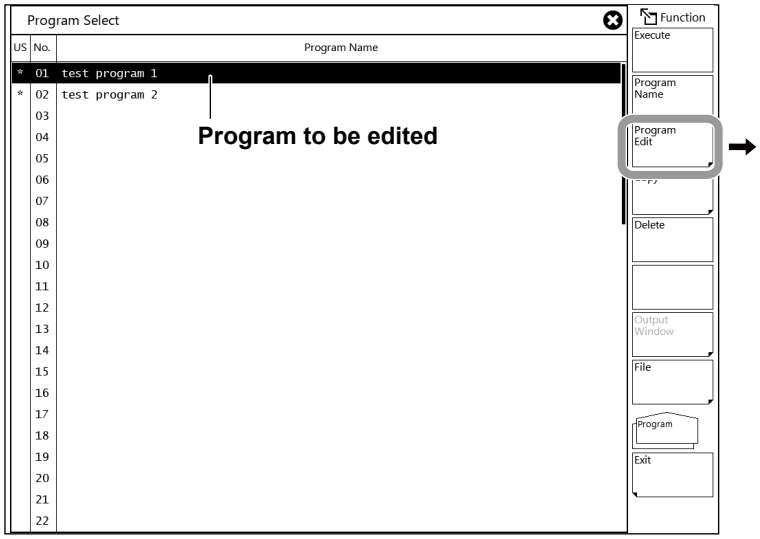
Note

You can also enter numbers and alphabetic characters directly from the numeric keypad of an external keyboard.

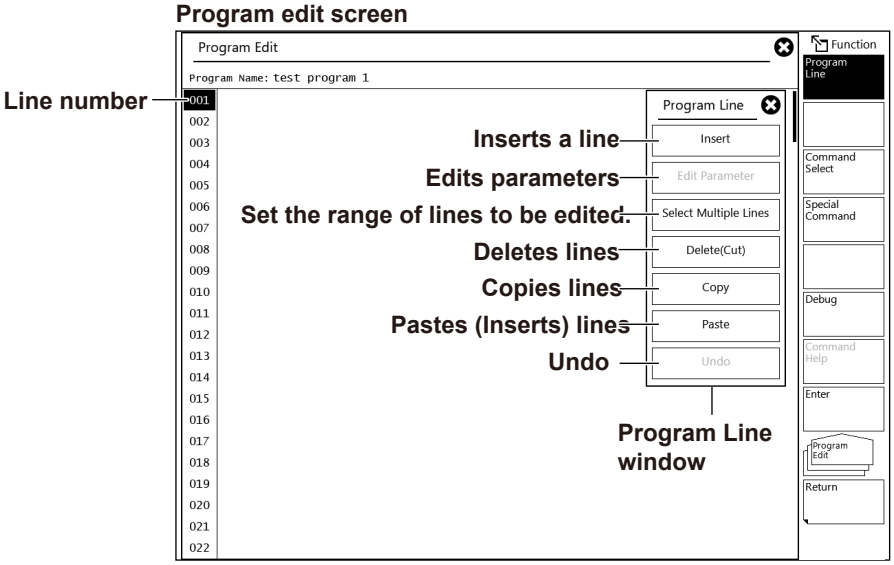
Editing a Program

8. On the program registration screen, select a program you want to edit, and tap **Program Edit**. A program edit screen appears.

Program registration screen



Program edit screen



For details on the Program Line window, see page 7-6 and subsequent pages.

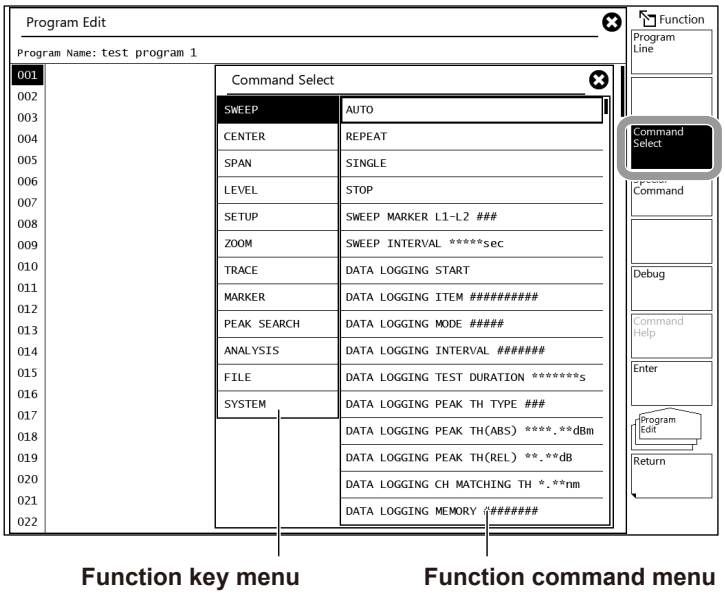
Selecting Commands

Function Commands

Function commands execute the same functions as the function switches (including the function menu contents). (Commands that correspond to Single, Span, and other function menus)

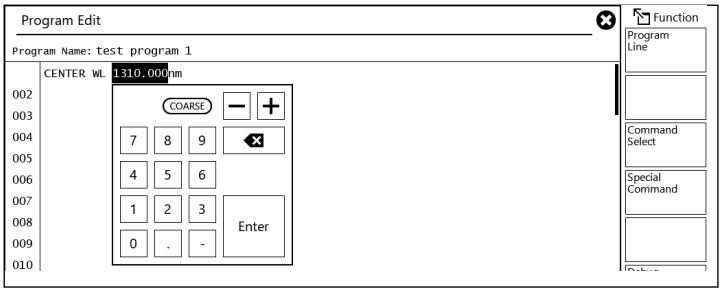
1. Tap **Command Select**. A function command selection screen appears.

Function commands



2. On the Command Select screen, tap a function command. The selected function command is entered. If you entered a command that requires a parameter to be set, a parameter edit screen appears.

Parameter edit screen (e.g., center wavelength)



Note

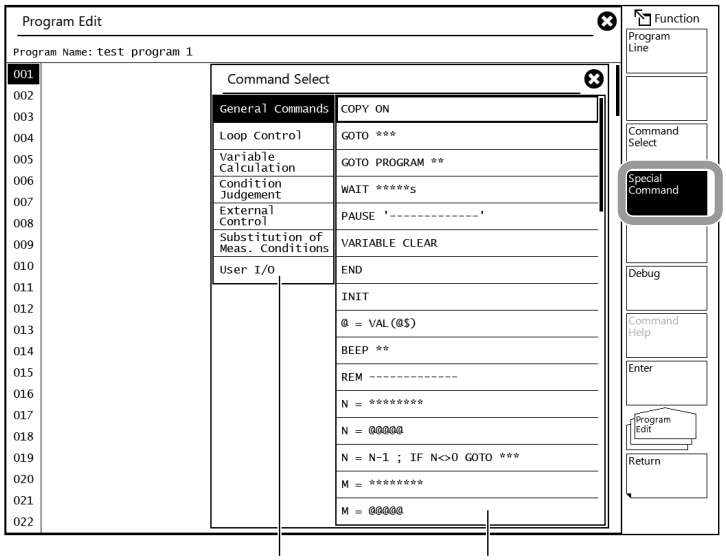
- If you tap a function key, a function command corresponding to the function key appears automatically in the top area.
- You can also select a special command using the rotary knob or arrow keys and tap enter.
- The "*****" area of the command is a value, the "###" area is the selected parameter, and the "- - - - - -" area is the text input area.
- Function commands can also be set by using the mouse. When you right-click, a panel key list appears. A function command corresponding to the panel key you click is entered.

Special Commands

These commands include jump commands, program control commands for conditional decision, etc., control commands to an external device, and data output commands.

1. Tap the **Special Command**. The special command selection screen is displayed.

Special commands

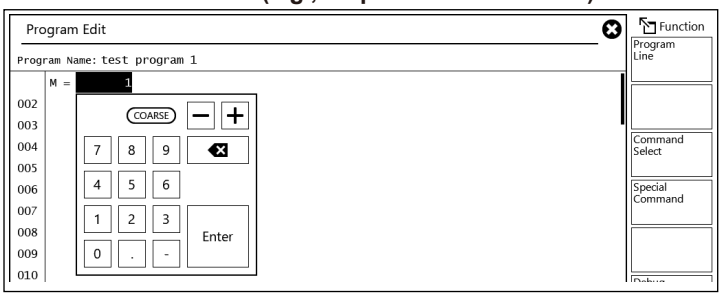


Jump commands and conditional decision menu

Special commands menu

2. On the Command Select screen, tap a special command. The selected special command is entered. If you entered a command that requires a parameter to be set, a parameter edit screen appears.

Parameter edit screen (e.g., loop control variable M)



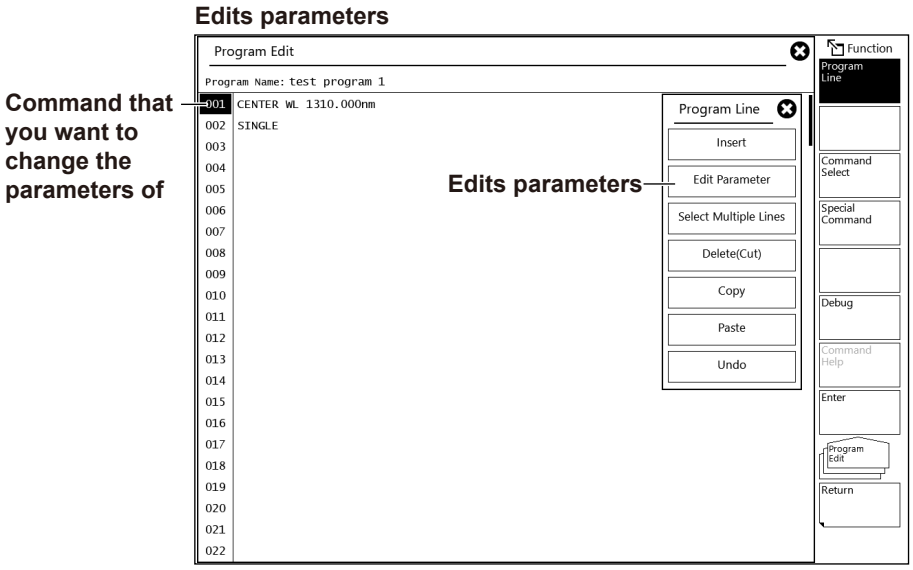
Note

- If you tap a jump command or conditional decision, a special command corresponding to the condition appears automatically in the top area.
- You can also select a special command using the rotary knob or arrow keys and tap enter.
- The "*****" area of the command is a value, the "###" area is the selected parameter, and the "- - - - - -" area is the text input area.
- Function commands can also be set by using the mouse. When you right-click, a panel key list appears. A function command corresponding to the panel key you click is entered.

Editing Parameters

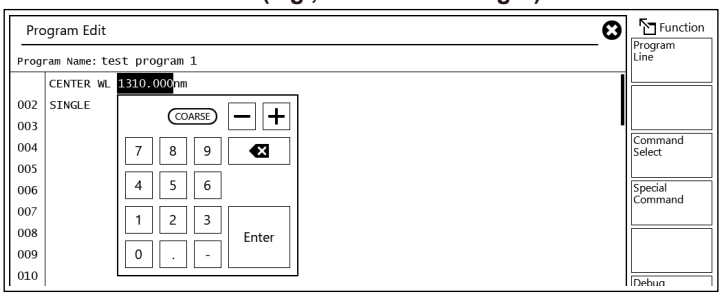
You can change the parameters of an entered command.

1. Tap the line containing the command that you want to change the parameter of. Edit Parameter in the Program Line window is enabled.



2. Tap **Edit Parameter** in the Program Line window. A parameter edit screen appears.

Parameter edit screen (e.g., center wavelength)



3. Enter a parameter, and tap **Enter** on the parameter edit screen. The parameter is set.

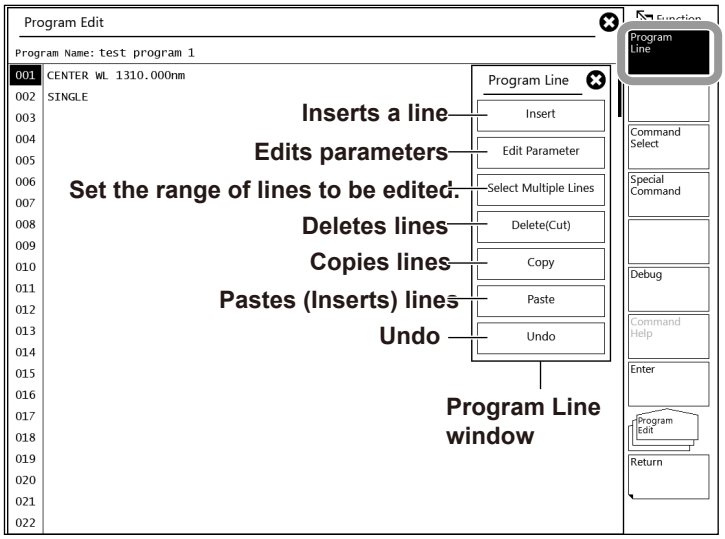
Note

The parameter edit screen that appears varies depending on the type of parameter.

Inserting, Copying, or Deleting a Line

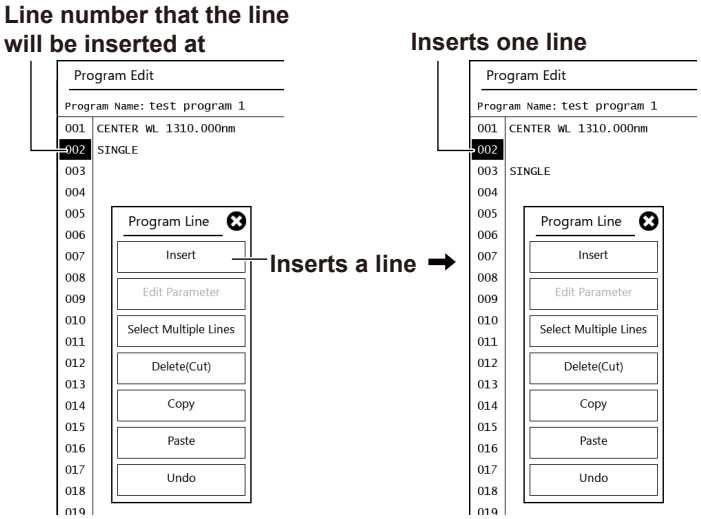
You can copy or delete the contents of a line.

1. Tap **Program Line**. A Program Line window appears.



Inserts a line

2. Tap the line number where you want to insert a line.
3. Tap **Insert** in the Program Line window. A line is inserted above the selected line number.

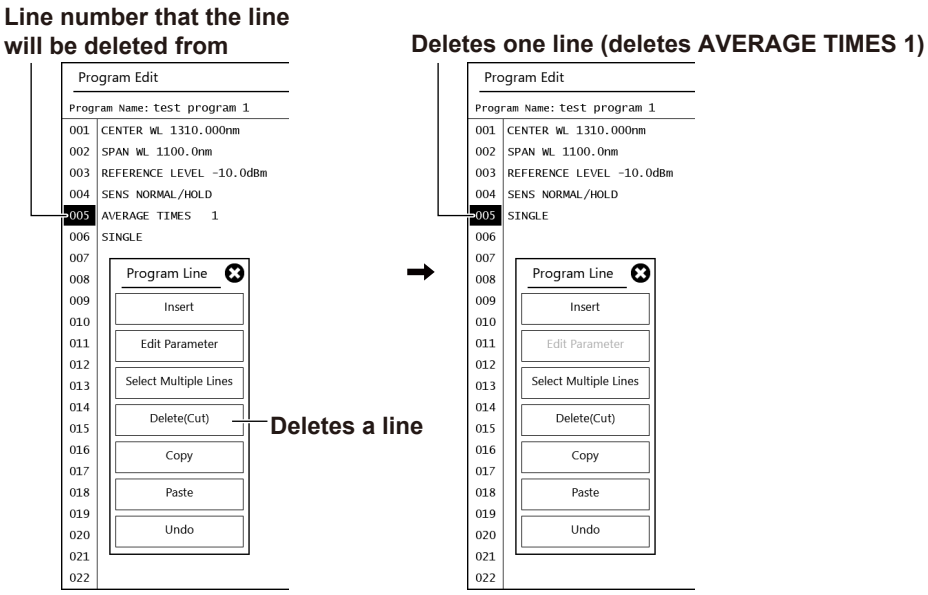


Note

Insertion is not possible when there is a command entered at line 200.

Deletes a line

2. Tap the line you want to delete.
3. Tap **Delete (Cut)** in the Program Line window. The selected line is deleted. To undo a deleted line, tap Undo in the Program Line window.

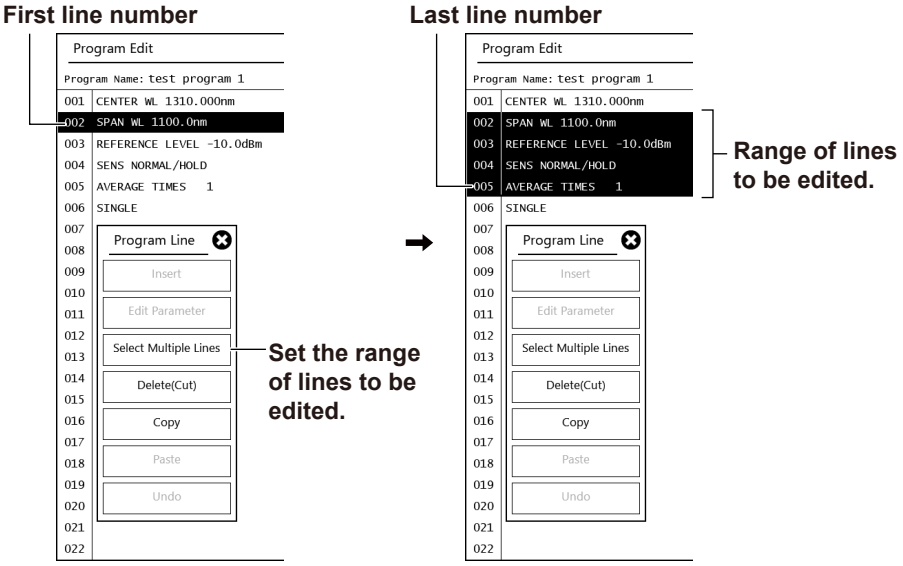


Note

To delete multiple lines, set the range of lines to be edited.

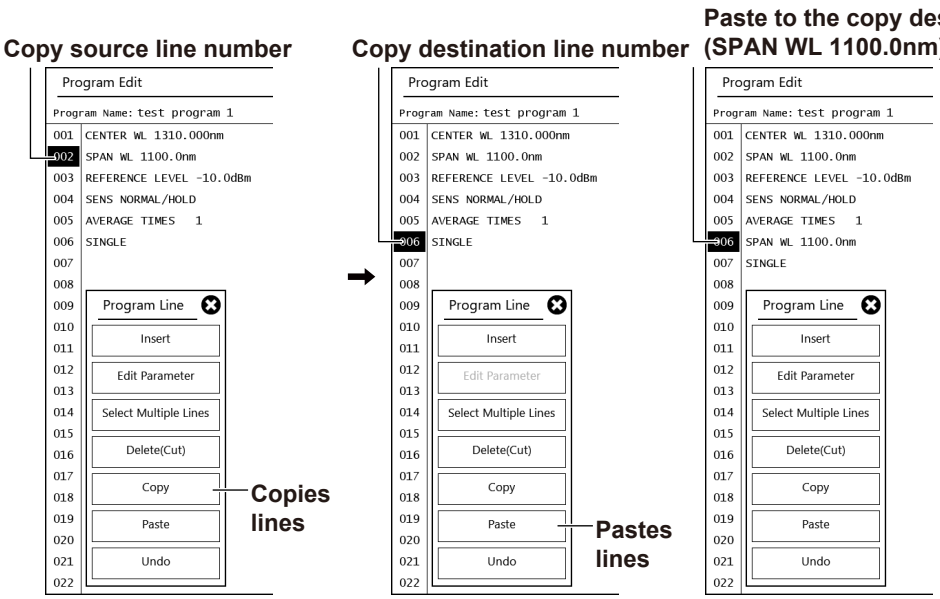
Set the range of lines to be edited.

2. Tap the first line number.
3. Tap **Select Multiple Lines** in the Program Line window.
4. Select the last line number. The edit range is set. If you change the last line number, the edit range will also change. If you tap **Select Multiple Lines** in the Program Line window, the edit range will be cleared.



Copying a Line

2. Tap a copy source line.
3. Tap **Copy** in the Program Line window.
4. Tap a copy destination line.
5. Tap **Paste** in the Program Line window. The copied content is inserted above the copy destination line you selected. To undo a copied line, tap **Undo** in the Program Line window.



Note

To copy multiple lines, set the range of lines to be edited.

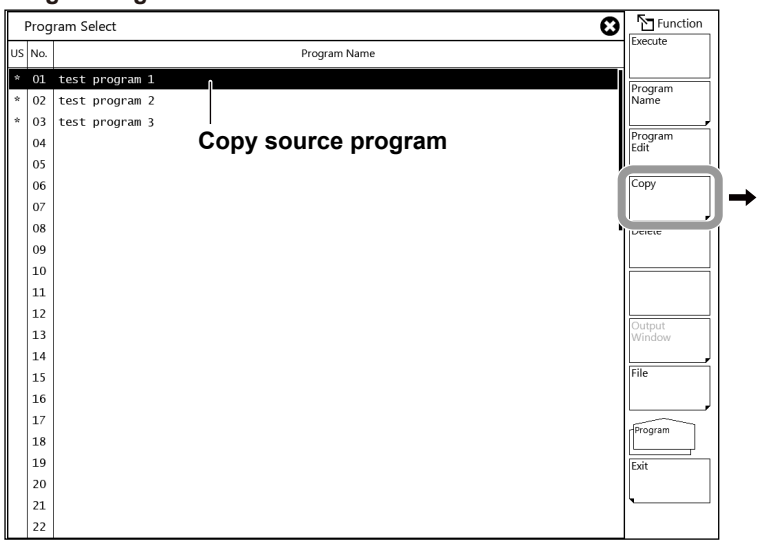
Copying a Program

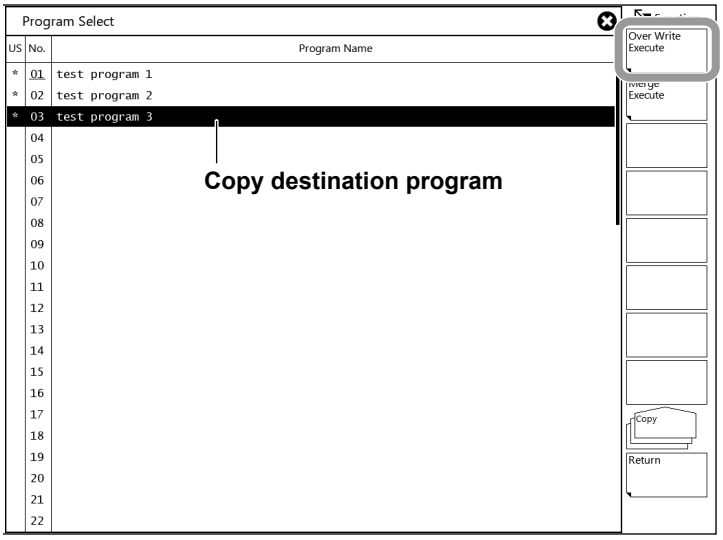
1. Press **APP**. The APP menu appears.
2. Double-tap **Program**, or tap the **Program** icon and then tap **Execute**. A program registration screen appears.
3. On the registration screen, tap a copy source program number.
4. Tap **Copy**. A Copy menu appears.

Overwriting with the Copied Program

5. Select a copy destination program, and tap **Over Write Execute**. The selected program is overwritten with the copied program.

Program registration screen

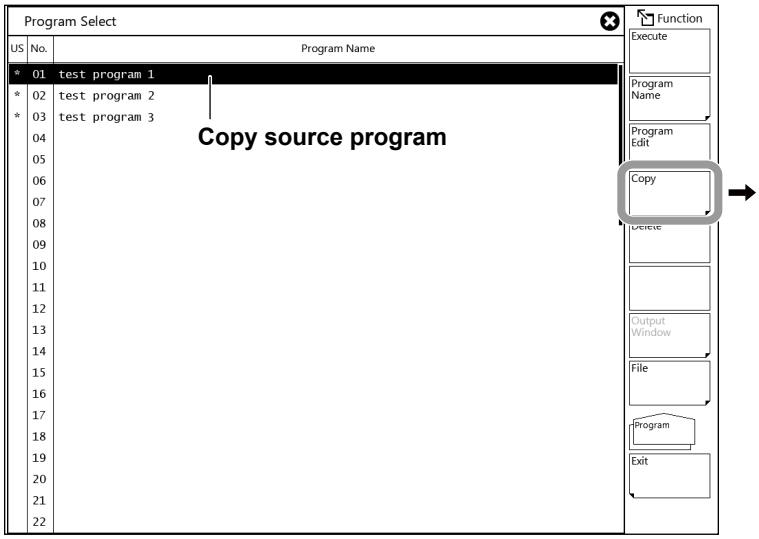


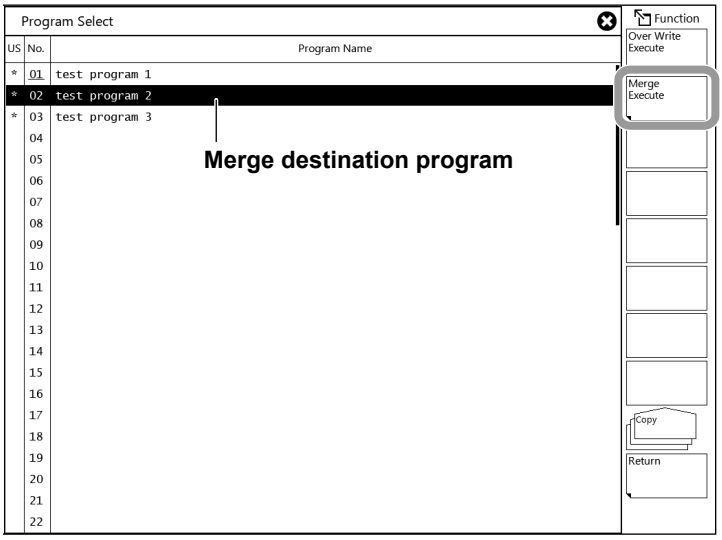


Merging the Copied Program

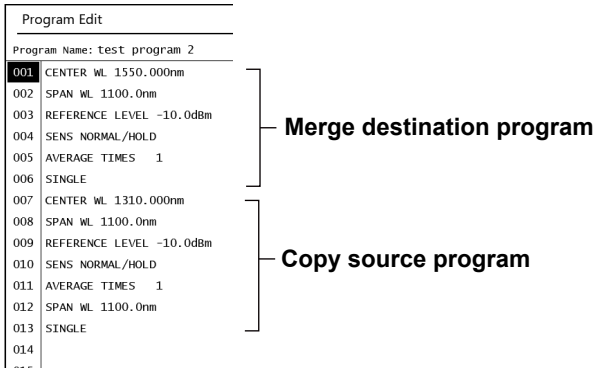
5. After step 4, select the copy destination program, and tap **Merge Execute**. The content of the copied program is inserted after the copy destination program.

Program registration screen





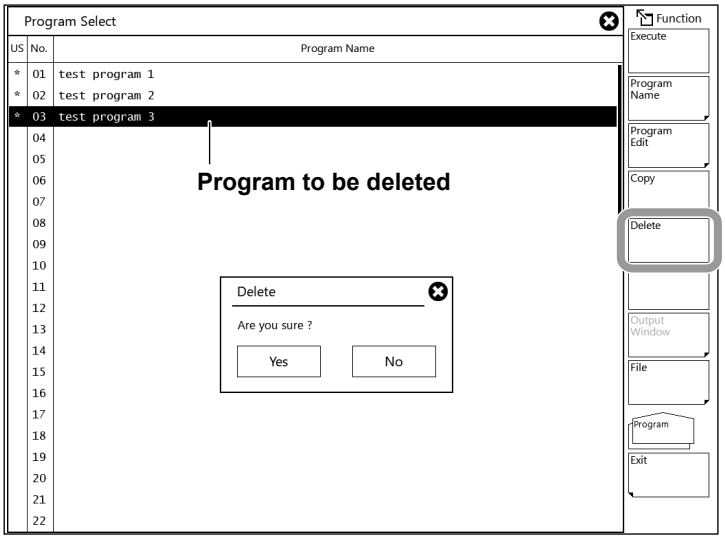
Program edit screen (test program 2)



Deleting a Program

1. Press **APP**. The APP menu appears.
2. Double-tap **Program**, or tap the **Program** icon and then tap **Execute**. A program registration screen appears.
3. On the registration screen, tap a program number you want to delete.
4. Tap **Delete**. A confirmation message appears.
5. To delete, tap **Yes**. To cancel, tap **No**.

Program registration screen



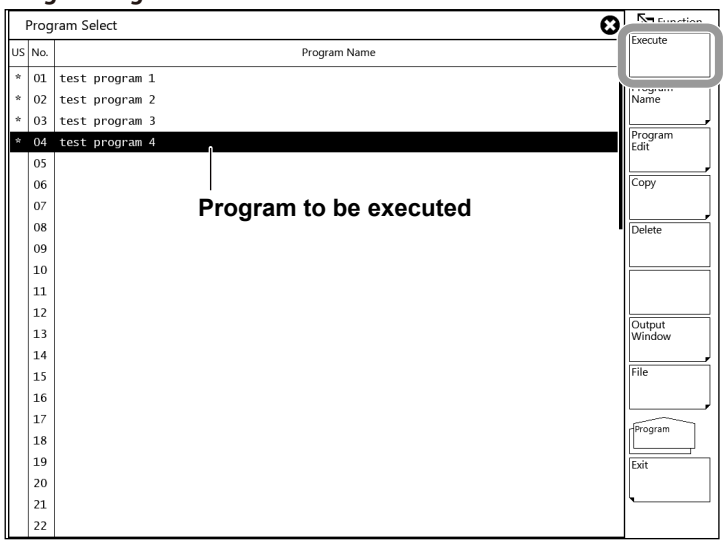
7.2 Executing a Program

Procedure

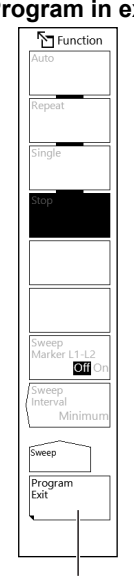
Executing a Program

1. Press **APP**. The APP menu appears.
2. Double-tap **Program**, or tap the **Program** icon and then tap **Execute**. A program registration screen appears.
3. On the registration screen, tap a program number you want to execute.
4. Tap **Execute**. The program is executed.

Program registration screen



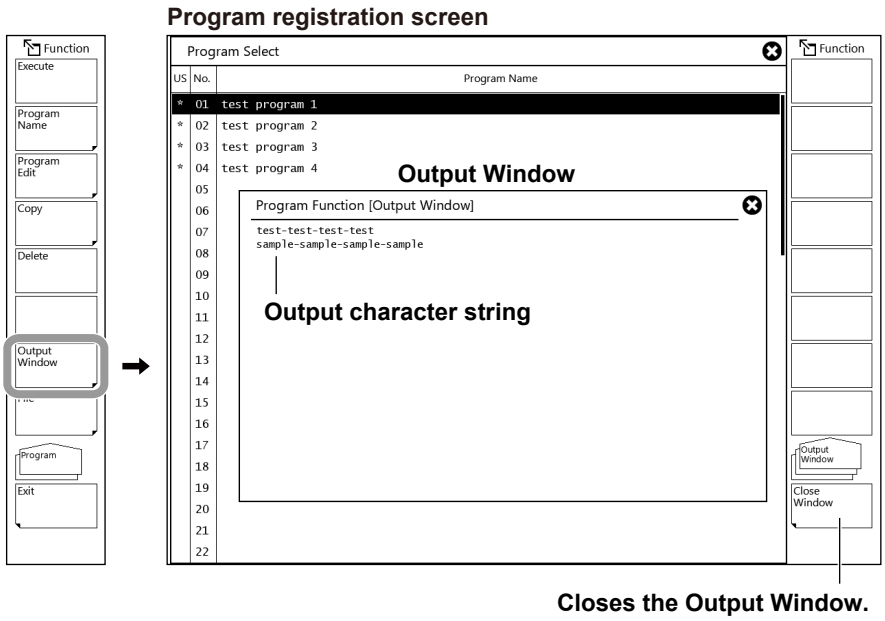
Program in execution



Tap to abort a running program.

OUTPUT WINDOW display

5. Tap **Output Window**. An Output Window appears.



Note

- If there is no data to be displayed in the Output Window, you cannot control it.
- The data or character that is output by executing a DATA OUTPUT command is displayed.
- The contents of the Output Window are held until execution of the OUTPUT WINDOW CLEAR special command.
- The contents of the Output Window can be stored in a file. See section 7.3.
- If the contents of the Output Window exceeds 200 lines, data will be erased beginning from the first line, in turn.
- Turning off the power switch on the instrument erases data in the Output Window.

Explanation

Using Special Commands

During program execution, you can perform unique operations with commands.

When Executing a Program Including "PAUSE '----------'"

The program pauses.

The message included in the "PAUSE '----------'" line is displayed, and the program pauses. To resume execution of the program, tap the Continue.

If a program is executed via remote control, the "PAUSE '----------' command is ignored.

When Executing a Program Including the "DATA INPUT ----------';@" Command

After the program executes, a data entry window is displayed.

In this case, one of two types of windows will appear depending on the @ variable.

String variables: Enter a file name using the same procedure as that of label input and tap the DONE.

Numerical variables: A data entry window is displayed. Enter an arbitrary number using the rotary knob, arrow keys, or ten key.

If a program is executed via remote control, the "DATA INPUT '---------- ';@ command is ignored.

Outputting Data Using "DATA OUTPUT @@@@@"

When executing a program, the OUTPUT WINDOW for displaying output data is displayed.

The contents of the variables specified by "@@@@@" appear in the OUTPUT WINDOW. Up to 200 lines can be displayed in the OUTPUT WINDOW. Only 20 lines can be displayed at once. To display lines other than the first 20, use the rotary knob or arrow keys to scroll.

The OUTPUT WINDOW can be displayed during execution of a program. To do so, use the "OUTPUT WINDOW ###" special command.

Note that the OUTPUT WINDOW disappears if the program ends.

Error Encountered upon Execution of a Program

If an error occurs during the running of a program, an error number indicating the details of the error is displayed in a window, and execution of the program is stopped.

Classification of Error Numbers

300–307 Errors caused by attempting a setting in manual operation which is disabled
320–326 Special command-related errors
340–347 Input/output-related errors
360–369 External memory-related errors
380, 381 Other errors

The above numbers can be read out using the SYSTem:ERRor[:NEXT]? command (see section 5.11, “SYSTem Group”).

No.	Message	Cause
300	Parameter out of range	A variable value is out of range or is not defined for a command that sets a parameter using variables.
302	Scale unit mismatch	There is a difference between the Y-axis scale of the active trace and the unit of a parameter in the “LINE MKR 3 or 4” command.
303	No data in active trace	Setting of the moving marker, a peak (or bottom) search, or activation of the analysis function was made with no data in the active trace.
304	Marker value out of range	Specified wavelength was out of the sweep range in the moving marker or line wavelength marker setting command.
305	No data in traces A or B	No waveform data in traces A or B when executing the “EDFA NF” command.
306	Invalid data	Trace had no data when attempting to save it to memory or to write it to FD/INT.
307	Unsuitable Write item	All data items were OFF at execution of “WRITE DATA”.
320	Undefined variable	A command containing an undefined variable was executed.
321	Variable unit mismatch	The unit of each variable does not agree within a command containing two or more variables.
322	Overflow	An overflow occurred in an arithmetic operation.
323	Undefined marker variable	A command containing a marker-value variable was executed when no marker had been displayed.
324	Invalid marker variable	A command containing the corresponding variable was executed at a time other than immedia tely after execution of a spectrum width search, peak search, etc.
325	Undefined line number	GOTO command’s jumping destination is a number other than 1 to 200.
326	F1 greater than F2	F1>F2 when the “IF F1 @@@@@ F2” command was executed.
345	Option does not respond	No response from an external device.
346	Option is not connected	No external device is connected.
360	Disk full	No file can be created due to insufficient free space in the external memory.
361	Disk not inserted	No external memory is connected.
362	Disk is write protected	The external memory is write protected.
363	Disk not initialized	External memory is not initialized. Or, it has been formatted in a format not supported by this instrument.
364	Directory full	Directory is full, therefore no file can be created.
365	File not found	The specified file cannot be read because it has not been found. Or, the file does not exist on the disk.
366	File is write protected	The file is specified to be read only, so that it cannot be rewritten or deleted.
367	No data	No data to store.
368	File is not a trace file	A file cannot be read because it is not a trace file.

No.	Message	Cause
369	Illegal file name	A file cannot be saved due to an incorrect file name.
371	Directory already exist	Unable to make the directory because a directory with the same name already exists.
380	Undefined program	An attempt was made to run a program that is not defined.
381	Syntax error	Command incorrect. (a program has been rewritten for some reason)
382	Program nesting over	Nesting is not possible because the program nesting is already too deep.
383	Program reentrant error	The destination of the GOTO command is set to its own program number. This will create an infinite loop.

7.3 Saving/Loading Program Data

Procedure

Programs created with the program function can be stored in the instrument's internal memory or a USB storage device. Stored programs can be loaded. For details on USB storage devices, see section 7.1 in IM AQ6374E-01EN. For details on file operation, see section 7.7 in IM AQ6374E-01EN.

CAUTION

Do not remove the USB storage medium or turn off the power when the USB storage media access indicator is blinking. Doing so may damage the USB storage medium or corrupt its data. Before you remove a USB storage medium, be sure to follow the procedure in section 6.1 of IM AQ6374E-01EN to make the USB storage medium ready to be removed.

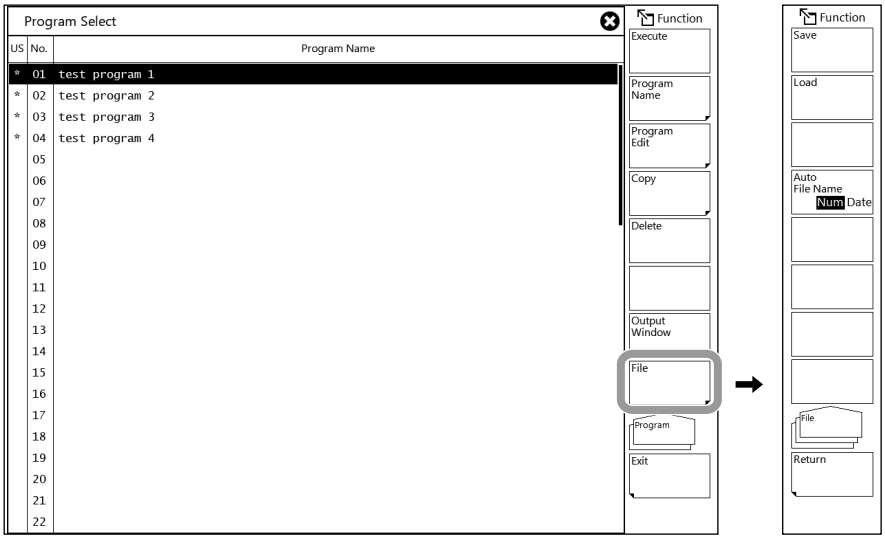
French

ATTENTION

Ne retirez pas le support de stockage USB et ne mettez pas l'alimentation hors tension lorsque l'indicateur d'accès au support de stockage USB clignote. Cela pourrait endommager le support de stockage USB ou corrompre ses données.
Avant de retirer un support de stockage USB, veuillez à suivre la procédure décrite dans la section 6.1 de l'IM AQ6374E-01EN pour rendre le support de stockage USB prêt à être retiré.

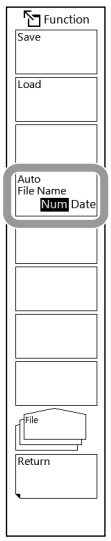
1. Press **APP**. The APP menu appears.
2. Double-tap **Program**, or tap the **Program** icon and then tap **Execute**. A program registration screen appears.
3. Tap **File**. A File menu appears.

Program registration screen



Selecting the Format of Files Saved Automatically with File Names

4. Tap **Auto File Name**. Select Num (serial number) or Date (date).

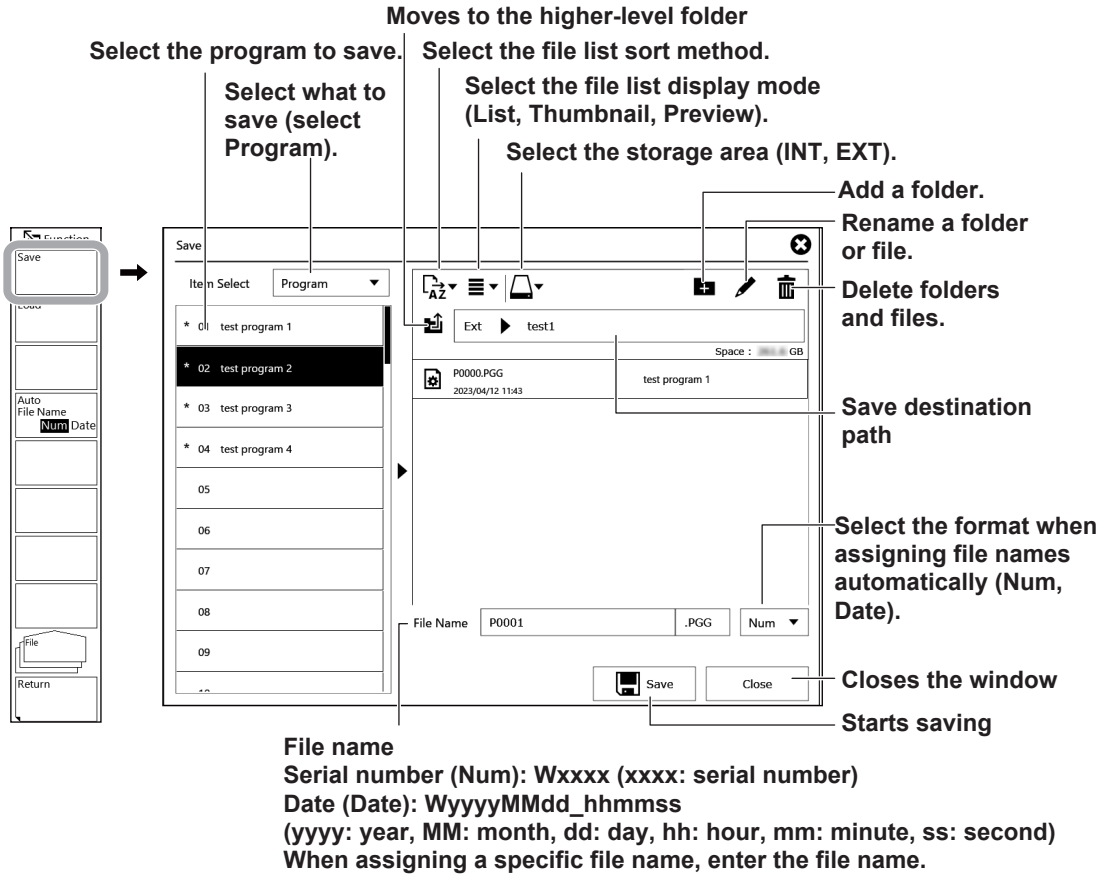


Note

You can also set this on the Save window, described later.

Saving Program Data

5. Tap **Save**. A Save window appears.
6. Tap the program data you want to save.
7. In the file list, open the folder you want to save program data in.



File name
Serial number (Num): Wxxxx (xxxx: serial number)
Date (Date): WyyyyMMdd_hhmmss
(yyyy: year, MM: month, dd: day, hh: hour, mm: minute, ss: second)
When assigning a specific file name, enter the file name.

Specifying a file name

You can save program data by specifying a file name. If you do not specify a file name, the file will automatically be assigned a name using a serial number or date.

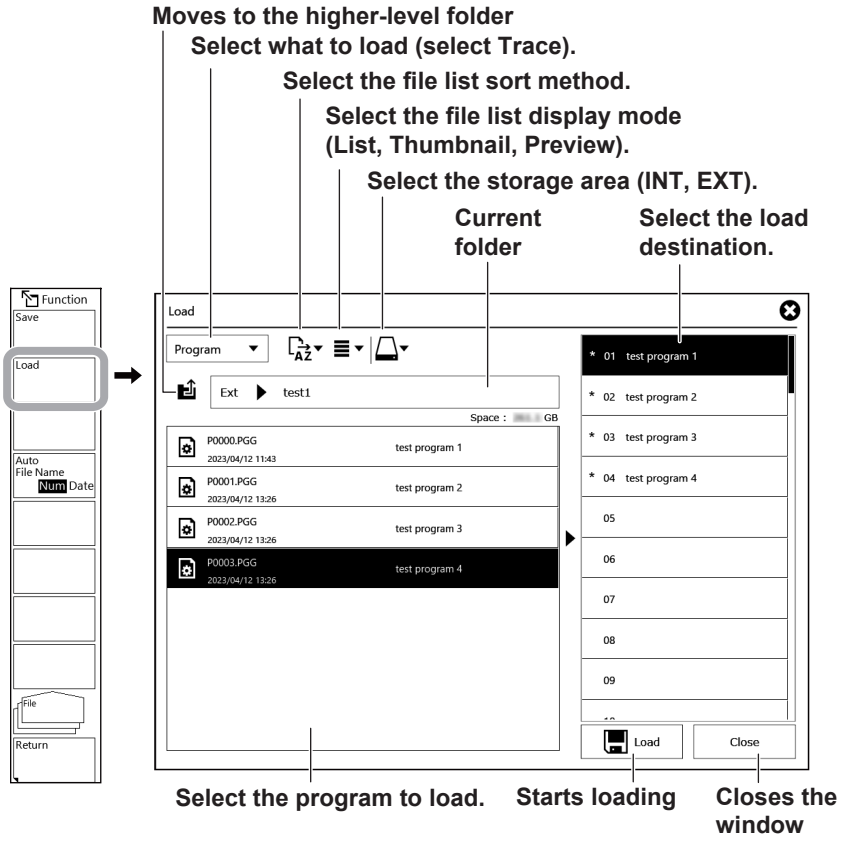
9. In the Save window, tap the file name display area twice. A keyboard appears.
10. Enter a file name using the keyboard, and tap Done. The file name will be entered.

Saving the file

11. On the Save window, tap **Save**. The file is saved. If you tap Close before you tap Save, the window will close without saving the data.
12. When overwriting a file, a confirmation message appears. Tap **OK**. To cancel overwriting, tap **Cancel**.

Loading Program Data

5. Tap **Load**. A Load window appears.



Loading the File

5. On the Load window, tap **Load**. The file is loaded.

If you tap Close before you tap Load, the window will close without loading the program.

If you select a program number that already has program data loaded and execute the load, the data will be overwritten with the new program data.

Explanation

Extension

The extension used when saving is as follows.
BIN (binary format): .PGG

File Name

You can have a file name automatically assigned, or specify an arbitrary name for the save. If you do not assign a file name, a file name is automatically assigned as follows.

When Auto File Name is set to NUM

File name: PXXXX.PGG
XXXX is a serial number from 0000 to 9999

When Auto File Name is set to DATE

File name: Wyyyymmdd_hhmmss.PGG
yyyymmdd: Year (Gregorian) month day
hhmmss: Hour (24 hours) minute second
(Date/time when the file list was updated)

Note

- The date/time used when file names are assigned based on the date through remote commands or program commands is the date/time when the file is created.
- Only use the characters allowed in file names by MS-DOS when changing a file name. The maximum file name length is 56 characters (including the extension). The following characters can be used in file names.
  !#$%&’()-
  0123456789@
  ABCDEFGHIJKLMNOPQRSTUVWXYZ^
  abcdefghijklmnopqrstuvwxyz{}¯

File Size

The file size is about 13 KB.

7.4 Program Function Commands

There are two types of commands used in a program: function commands which are input using a panel switch, and other special commands.

Variables

Variables that can be used in a program are shown in the table below.

Type	Variable Name	Description
Generalized variables	E, G–K, O–V, X–Z	Contains a generalized numeric variable.
	A$, B$, C$, D$	Contains a generalized string variable.
Special characters variables	FILE$	Contains the name of the last file accessed.
	TIME$	Contains the date and time. (Ex. 2023 Aug 08 20:45:37)
Marker variables	WM	Contains the wavelength value of the moving marker.
	W1	Contains the wavelength value of fixed marker 1.
	W2	Contains the wavelength value of fixed marker 2.
	W2-W1	Contains the wavelength difference between fixed markers 1 and 2.
	W(CH)	Contains the level values of fixed markers (CH: 1 to 1024).
	LM	Contains the level value of the moving marker.
	L1	Contains the level value of fixed marker 1.
	L2	Contains the level value of fixed marker 2.
	L2-L1	Contains the level difference between fixed markers 1 and 2.
	L(CH)	Contains the level values of fixed markers (CH: 1 to 1024).
Analysis variables	SPWD	Contains spectrum width applied in making a spectrum width search.
	MEANWL	Contains center wavelength applied in making spectrum width search.
	PKWL	Contains a peak (or bottom) wavelength value applied in making a peak (or bottom) search or spectrum width search.
	PKLVL	Contains a peak (or bottom) level value applied in making a peak (or bottom) search or spectrum width search.
	MODN	Contains the number of modes applied in making a spectrum width search.
	SMSR	Contains the side mode suppression ratio (level difference) applied in making SMSR measurements.
	SMSR2	Contains the longer wavelength side's value of the side mode suppression ratio (level difference) when an SMSR3 or SMSR4 measurement is executed.
	WDMCHN	Contains the number of channels detected in performing WDM analysis.
	WDMWL(CH)	Contains the center wavelength of channel CH used in performing WDM analysis.
	WDMLVL(CH)	Contains the level of channel CH used in performing WDM analysis.
	WDMSMSR(CH)	Contains SNR of channel CH used in performing WDM analysis.
	NFCHN	Contains the number of channels detected in performing EDFA-NF analysis.
	NFWL(CH)	Contains the center wavelength of channel CH used in performing EDFA-NF analysis.
	NFLVL(CH)	Contains the input signal level of channel CH used in performing EDFA-NF analysis.
	NFLVLO(CH)	Contains the output signal level of channel CH used in performing EDFA-NF analysis.
	NFASELV(CH)	Contains the ASE level of channel CH used in performing EDFA-NF analysis.
	NFGAIN(CH)	Contains the gain of channel CH used in performing EDFA-NF analysis.

Type	Variable Name	Description
Analysis variables	NFNF(CH)	Contains NF of channel CH used in performing EDFANF analysis.
	MKPWR	Contains power obtained in making between line-markers power measurements.
	DOMWL	Dominant wavelength value is entered when measuring the dominant wavelength.
	XCOL	The chromaticity coordinate value X is entered when perfoming COLOR analysis.
	YCOL	The chromaticity coordinate value Y is entered when perfoming COLOR analysis.
	ZCOL	The chromaticity coordinate value Z is entered when perfoming COLOR analysis.
Program control variables	M	Contains loop counter data.
	N	Contains loop counter data.
	F1	Contains a conditional judgment variable.
	F2	Contains a conditional judgment variable.
	CH	Contains an element number variable used in accessing an array variable (1–1024).

Principles of Variable-based Arithmetic Operations

For assignment of units after arithmetic operations when a variable with a unit is used in the operation, see below.

Expression	Results
(With a unit) × (Without unit)	With a unit
(With a unit)/(Without unit)	With a unit
(Without unit) + (Without unit)	Without unit
(Without unit) – (Without unit)	Without unit
(Without unit) × (Without unit)	Without unit
(Without unit) / (Without unit)	Without unit
(nm) + (nm)	(nm)
(nm) – (nm)	(nm)
(nm) / (nm)	Without unit
(dB) + (dB)	(dB)
(dB) – (dB)	(dB)
(dBm) + (dB)	(dBm)
(dBm) – (dB)	(dBm)
(dBm) – (dBm)	(dB)
(#W) + (#W)	(#W)
(#W) – (#W)	(#W)
(#W) / (#W)	Without unit

Note

- For the units of dBm/nm, W/nm, dB/km, and %, dBm, W, dB, and without unit apply respectively in terms of variables.
- Arithmetic operations are made as noted above according to the unit of a variable, and the unit is appended to the result obtained after operation.
- If an arithmetic operation is made in any combination other than the above (addition, subtraction, multiplication, or division of variables with different units), the result of the operation has no units.
- The units of #W are treated as follows:
  1 mW=1
  1 mW=0.001
  1 nW=0.000001
  1 pW=0.000000001

Specifications of “@=VAL(@$)” Command

A character string other than the numerics located before a value (starting with a sign or number) in @$ character string will be ignored, and are converted as follows.

- “ , ”
- Numbers up to the next string or delimiter

If no numeric exists in @$ character string, “0” is substituted for variable @.

List of Function Commands

A description is given of the program commands of each function command. The optical spectrum analyzers on which the program command is valid is indicated along with the parameter range and variables supported.

SWEEP

Program Command	Description	Parameter ranges and supported variables.
AUTO	Auto sweep	
REPEAT	Repeat sweep	
SINGLE	Single sweep	
STOP	Sweep stop	
SWEEP MARKER L1-L2 ###	Selects ON/OFF of sweep function between markers	###: ON or OFF
SWEEP INTERVAL *****sec	Sets the interval time for repeat sweep	MINIMUM, 1 to 99999 sec (1 step) (MINIMUM when set to 0.)
DATA LOGGING START	Starts data logging	
DATA LOGGING ITEM ########	Sets the data logging source	########: WDM, PEAK, MULTIPEAK, DFBLD
DATA LOGGING MODE ####	Sets the data logging mode (maximum channel mode or maximum logging mode)	MODE1 (MAX 1024 ch, 2001 times), MODE2 (MAX 256 ch, 10001 times)
DATA LOGGING INTERVAL #######	Sets the measurement interval of data logging	####: SWEEP TIME, 1 sec, 2 sec, 5 sec, 10 sec, 30 sec, 1 min, 2 min, 5 min, 10 min
DATA LOGGING TEST DURATION *******s	Sets the measurement duration of data logging (in seconds)	*******: 1 to 8639999 (1 step)
DATA LOGGING PEAK TH TYPE ###	Sets how the threshold for detecting the data logging mode (peak or bottom) is specified	###: ABS, REL
DATA LOGGING PEAK TH(ABS) ****.**dBm	Sets the threshold (absolute value) for detecting the data logging mode	****.**: 20.00 to -100.00
DATA LOGGING PEAK TH(REL) **.**dB	Sets the threshold (relative value) for detecting the data logging mode	**.**: 0.01 to 99.99
DATA LOGGING CH MATCHING TH ± *.**nm	Sets the threshold of the channel-matching wavelength λ for data logging	*.**: 0.01 to 1.00
DATA LOGGING MEMORY ########	Sets the temporary area for saving waveform files of data logging	###: INTERNAL, EXTERNAL
DATA LOGGING TRACE LOGGING ###	Sets whether waveforms will be logged during data logging	###: ON/OFF

CENTER

Program Command	Description	Parameter ranges and supported variables.
CENTER WL ****.***nm	Sets measurement center wavelength.	350.000 to 1750.000 nm (0.001 step)
CENTER WL @@@@@	Sets the value of variable @@@@@ to measurement center wavelength.	@@@@@: E, G, H, I, J, K, O, P, Q, R, S, T, U, V, X, Y, Z, WM, W1, W2, W(CH), DOMWL, PKWL, MEANWL, WDMWL(CH), NFWL(CH)
CENTER FREQ ***.****THz	Sets measurement center frequency.	171.0000 to 857.0000 THz (0.0001 step)
CENTER FREQ @@@@@	Sets the value of variable @@@@@ to measurement center frequency.	@@@@@: E, G, H, I, J, K, O, P, Q, R, S, T, U, V, X, Y, Z, WM, W1, W2,WAM1, WAM2, WAM3, WAM4, W(CH), PKWL, DOMWL, MEANWL, WDMWL(CH), NFWL(CH)
CENTER WNUM ****.***cm^{-1}	Sets measurement center wavenumber.	5714.000 to 28571.000 cm^{-1} (0.001 step)
CENTER WNUM @@@@@	Sets the value of variable @@@@@ to measurement center wavenumber.	@@@@@: E, G, H, I, J, K, O, P, Q, R, S, T, U, V, X, Y, Z, WM, W1, W2,WAM1, WAM2, WAM3, WAM4, W(CH), PKWL, DOMWL, MEANWL, WDMWL(CH), NFWL(CH)
START WL ****.***nm	Sets measurement-starting wavelength.	1.000 to 1750.000 nm (0.001 step)
START FREQ ***.****THz	Sets measurement-starting frequency.	10.0000 to 857.0000 THz (0.0001 step)
START WNUM ****.***cm^{-1}	Sets measurement-starting wavenumber.	1000.000 to 28571.000 cm^{-1} (0.001 step)
STOP WL ****.***nm	Sets measurement-ending wavelength.	350.000 to 2450.000nm (0.001 step)
STOP FREQ ***.****THz	Sets measurement-ending frequency.	171.0000 to 999.9000 THz (0.0001 step)
STOP WNUM ****.***cm^{-1}	Sets measurement-ending wavenumber.	5714.000 to 40000.000 cm-1 (0.001 step)
PEAK->CENTER	Sets the center frequency of the waveform on the active trace.	
MEAN WL->CENTER	Performs a spectrum width search on the active trace, and sets the results of center wavelength to the measurement center wavelength.	
AUTO CENTER ###	Executes every time a sweep finishes. <PEAK →CENTER> Function ON/Selects OFF	
VIEW SCALE->MEASSCALE	Sets the current display conditions to measuring conditions.	

SPAN

Program Command	Description	Parameter ranges and supported variables.
SPAN WL ****.*nm	Sets the measuring span.	0.0 to 1100.0 nm (0.1 step)
SPAN WL @@@@@	Sets the value of variable @@@@@ to the measuring span.	@@@@@: E, G, H, I, J, K, O, P, Q, R,S, T, U, V, X, Y, Z, W2-W1, SPWD
SPAN FREQ ***.**THz	Sets the measuring span.	0.0 to 330.00 THz (0.01 step)
SPAN FREQ @@@@@	Sets the value of variable @@@@@ to the measuring span.	@@@@@: E, G, H, I, J, K, O, P, Q, R,S, T, U, V, X, Y, Z, W2-W1, SPWD
SPAN WNUM ****.*cm^{-1}	Sets the measuring span.	0, 0.5 to 22858.000 cm^{-1} (0.001 step)
SPAN WNUM @@@@@	Sets the value of variable @@@@@ to the measuring span.	@@@@@: E, G, H, I, J, K, O, P, Q, R,S, T, U, V, X, Y, Z, W2-W1, SPWD
Δλ ->SPAN	Performs a spectrum width search on the active trace, and sets the results to the measuring span.	
0nm SWEEP TIME **sec	Sets sweep time used when a sweep is made in a 0 nm measuring span.	0(MINIMUM), 1 to 50 (1step)

LEVEL

Program Command	Description	Parameter ranges and supported variables.
REFERENCE LEVEL ***.*dBm	Sets the reference level valueused for LOG scaling.	–90.0 to 30.0dBm (0.1 step)
REFERENCE LEVEL ***.*##	Sets the reference level value used for LIN scaling.	1.00pW to 1000mW (1.00 to 9.99 [pW, nW, µW, mW] :0.01 step 10.0 to 99.9(100)[pW, nW, µW, (mW)] :0.1 step 100 to 999 [pW, nW, µW, mW] : 1 step) ## is , pW, nW, µW, mW (select one of the above)
REFERENCE LEVEL @@@@@	Sets the value of variable @@@@@ to the reference level value.	@@@@@: E, G, H, I, J, K, O, P, Q, R,S, T, U, V, X, Y, Z, LM, L1, L2, L(CH), PKLVL, WDMLVL(CH), NFLVI(CH), NFLVO(CH), NFASELV(CH), MKPWR
LEVEL SCALE **.*dB/D	Sets a level scale value.	0 (LINEAR), 0.1 to 10.0dB/DIV (0.1 step)
LEVEL SCALE @@@@@	Sets the value of variable @@@@@ to the level scale.	@@@@@: E, G, H, I, J, K, O, P, Q, R,S, T, U, V, X, Y, Z, L2-L1, SMSR, SMSR2, WDMSNR(CH), NFNF(CH)
BASE LEVEL ****	Lower value for linear scale setting. Use units set under REF LEVEL. If exceeds 90% of upper units of scale, execution error results.	0 to 900 (0.1 step)
PEAK->REF LEVEL	Sets peak level of the waveform on the active trace to the reference level value.	
AUTO REF LEVEL ###	Executes after each sweep finishes. Selects ON/OFF for the <PEAK → REF LEVEL> function.	###: ON/OFF
LEVEL UNIT ######	Sets the unit of a level scale.	###: dBm, dBm/nm
Y SCALE DIVISION ##DIV	Sets the level scale division.	##: 8, 10
Y MAIN SCALE INITIALIZE	Resets the main level scale to the initial state.	
REF LEVEL POSITION **DIV	Sets the position of the reference level on the level scale.	0 to 10 (1 step)
SUB SCALE LOG **.*dB/D	Sets the sub scale value used for LOG scaling.	0.1 to 10.0dB/DIV (0.1 step)

Program Command	Description	Parameter ranges and supported variables.
SUB SCALE LIN *.***/D	Sets the sub scale value used for LIN scaling.	0.005 to 1.250 (0.005 step)
SUB SCALE **.*dB/km	Sets the sub scale value used for dB/km scaling.	
SUB SCALE **.*%/D	Sets the sub scale value used for %D scaling.	0.5 to 125.0 (0.1 step)
OFFSET LEVEL **.*dB	Sets the sub scale offset value used for LOG scaling.	–99.9 to 99.9 (0.1 step)
SCALE MINIMUM **.**	Sets the lower sub scale value used for linear scaling.	000 to 12.50 (0.01 step)
OFFSET LEVEL ***.*dB/km	Sets the sub scale offset value used for dB/km scaling.	
SCALE MINIMUM ****.*%	Sets the lower sub scale value used for &D scaling.	0.0 to 1250.0 (0.1 step)
LENGTH **.***km	Sets fiber length.	0.001 to 99.999 (0.001 step)
AUTO SUB SCALE ###	Automatically sets the sub scale from the calculated trac e waveform.	###: ON/OFF
SUB REF LEVEL POSITION **DIV	Sets the position of the reference level on the sub level scale	0 to 10 (1 step)
Y SUB SCALE INITIALIZE	Resets the sub level scale to the initial state.	

SETUP

Program Command	Description	Parameter ranges and supported variables.
RESOLUTION WL *.****nm	Sets the wavelength resolution.	0.05 to 10.000 (1-2-5 step)
RESOLUTION WL @@@@@	Sets the value of variable @@@@@ to the wavelength resolution.	@@@@@: E, G, H, I, J, K, O, P, Q, R,S, T, U, V, X, Y, Z, W2-W1, SPWD
RESOLUTION FREQ ***GHz	Sets the frequency resolution.	10 to 2000 (1-2-4 step)
RESOLUTION FREQ@@@@@	Sets the value of variable @@@@@ to the frequency resolution.	@@@@@: E, G, H, I, J, K, O, P, Q, R,S, T, U, V, X, Y, Z, W2-W1, SPWD
RESOLUTION WNUM *.**cm^{-1}	Sets the wavenumber resolution.	0.1 to 25.000 (1-2.5-5 step)
RESOLUTION WNUM @@@@@	Sets the value of variable @@@@@ to the wavenumber resolution.	@@@@@: E, G, H, I, J, K, O, P, Q, R,S, T, U, V, X, Y, Z, W2-W1, SPWD
SENS NORMAL/HOLD	Sets the measuring sensitivity to NORMAL/HOLD.	
SENS NORMAL/AUTO	Sets the measuring sensitivity to NORMAL/AUTO.	
SENS NORMAL	Sets measuring sensitivity to NORMAL.	
SENS MID	Sets measuring sensitivity to MID.	
SENS HIGH1	Sets measuring sensitivity to HIGH1.	
SENS HIGH2	Sets measuring sensitivity to HIGH2.	
SENS HIGH3	Sets measuring sensitivity to HIGH3.	
SENSITIVITY LEVEL ***dBm	Sets the sensitivity level	
CHOPPER #####	Switches chopper mode.	#####: OFF/SWITCH
SWEEP SPEED ##	Sets the sweep speed.	##: 1x/2x
AVERAGE TIMES ***	Sets the number of averaging times.	1 to 999 (1 step)
AVERAGE TIMES @	Sets the number of averaging times to the value of variable.	@: E, G, H, I, J, K, O, P, Q, R,S, T, U, V, X, Y, Z, M, N
SAMPLING POINT AUTO ###	Sets sampling points per sweep automatically.	###: ON/OFF
SAMPLING POINT *****	Sets sampling points per sweep.	101 to 100001 (1 step)

Program Command	Description	Parameter ranges and supported variables.
SAMPLING POINT @	Sets the sampling points to the variable @.	@: E, G, H, I, J, K, O, P, Q, R,S, T, U, V, X, Y, Z, M, N
SAMPLING INTERVAL *.**** nm	Sets the measurement sampling interval per sweep.	0.001 to SPAN/101 (0.001 step)
SAMPLING INTERVAL @	Sets the sampline interval per sweep to the value of variable @.	@: E, G, H, I, J, K, O, P, Q, R,S, T, U, V, X, Y, Z, W2-W1, SPWD
X SCALE UNIT WAVELENGTH	Sets axis X to wavelength display mode.	
X SCALE UNIT FREQUENCY	Sets axis X to frequency display mode.	
X SCALE UNIT WAVENUMBER	Sets axis X to wavenumber display mode.	
PLS LIGHT MEASURE OFF	Turns OFF pulse light measurement mode.	
PEAK HOLD****msec	Sets the HOLD time for PEAK HOLD pulse light measurement.	****: 1 to 9999
EXTERNAL TRIGGER MODE	Sets external trigger mode.	
EXTERNAL TRIGGER EDGE RISE	Detects the falling edge of an external trigger signal.	
EXTERNAL TRIGGER EDGE FALL	Detects the rising edge of an external trigger signal.	
EXTERNAL TRIGGER DELAY ******µs	After detection of an external trigger signal, and sets the delay time until data acquisition.	******: 0 to 100000 (1 step)
GATE MODE ****.*msec	Sets the sampling interval on the gate sampling mode.	****.* : 0.1 to 1000.0 (0.1 step)
GATE LOGIC POSI	Sets the gate signal logic of gate sampling to the positive logic.	
GATE LOGIC NEGA	Sets the gate signal logic of gate sampling to the negative logic.	
MEASURE DELAY ****.*msec	Sets the measure delay on the gate sampling mode.	****.*: 0.1 to 1000.0 (0.1 step)
TRIGGER OUTPUT OFF	Turns OFF the trigger output mode.	
TRIGGER OUTPUT SWEEP STATUS	Sets the trigger output mode to sweep status.	
TRIGGER INPUT SAMPLING TRIGGER	Sets the trigger input mode tosampling trigger.	
TRIGGER INPUT SWEEP TRIGGER	Sets the trigger input mode to sweeptrigger.	
TRIGGER INPUT SAMPLING ENABLE	Sets the trigger input mode to sampling enable.	
RESOLUTION CORRECTION ###	Turns the wavelength resolution correction function ON/OFF.	###: ON/OFF
SMOOTHING ###	Turns the smoothing function ON/OFF.	###: ON/OFF
FIBER CORE SIZE #####	Switches the fiber core size mode.	#####: SMALL/LARGE

TRACE

Program Command	Description	Parameter ranges and supported variables.
ACTIVE TRACE #	Sets trace # to active trace.	#: A to G
DISPLAY #	Sets trace # to display mode.	#: A to G
BLANK #	Sets trace # to invisible mode.	#: A to G
WRITE #	Sets trace # to write mode.	#: A to G
FIX #	Sets trace # to data-fixing mode.	#: A to G
MAX HOLD #	Sets trace # to max. value detection mode.	#: A to G
MIN HOLD #	Sets trace # to min. value detection mode.	#: A to G
ROLL AVG # ***	Sets trace # to sequential addition averaging mode.	#: A to G, 2 to 100 (1 step)
C=A-B(LOG)	Sets trace C to TRACE A-B computation mode (LOG).	
C=B-A(LOG)	Sets trace C to TRACE B-A computation mode (LOG).	
C=A+B(LOG)	Sets trace C to TRACE A+B computation mode (LOG).	
C=A+B(LIN)	Sets trace C to TRACE A+B computation mode (LIN).	
C=A-B(LIN)	Sets trace C to TRACE A-B computation mode (LIN).	
C=B-A(LIN)	Sets trace C to TRACE B-A computation mode (LIN).	
C=1-k(A/B) k=*.****	Sets trace C to 1-k (TRACE A/B) computation mode.	1.0000 to 20000.0000 (0.0001 step)
C=1-k(B/A) k=*.****	Sets trace C to 1-k (TRACE B/A) computation mode.	1.0000 to 20000.0000 (0.0001 step)
F=C-D(LOG)	Sets trace F to TRACE C-D computation mode (LOG).	
F=D-C(LOG)	Sets trace F to TRACE D-C computation mode (LOG).	
F=C+D(LOG)	Sets trace F to TRACE C+D computation mode (LOG).	
F=D-E(LOG)	Sets trace F to TRACE D-E computation mode (LOG).	
F=E-D(LOG)	Sets trace F to TRACE E-D computation mode (LOG).	
F=D+E(LOG)	Sets trace F to TRACE D+E computation mode (LOG).	
F=C+D(LIN)	Sets trace F to TRACE C+D computation mode (LIN).	
F=C-D(LIN)	Sets trace F to TRACE C-D computation mode (LIN).	
F=D-C(LIN)	Sets trace F to TRACE D-C computation mode (LIN).	
F=D+E(LIN)	Sets trace F to TRACE D+E computation mode (LIN).	
F=D-E(LIN)	Sets trace F to TRACE D-E computation mode (LIN).	
F=E-D(LIN)	Sets trace F to TRACE E-D computation mode (LIN).	
F=POWER/NBW A **.*nm	Sets the power spectral density of trace A to be displayed on trace F.	
F=POWER/NBW B **.*nm	Sets the power spectral density of trace B to be displayed on trace F.	
F=POWER/NBW C **.*nm	Sets the power spectral density of trace C to be displayed on trace F.	

Program Command	Description	Parameter ranges and supported variables.
F=POWER/NBW D **.*nm	Sets the power spectral density of trace D to be displayed on trace F.	
F=POWER/NBW E **.*nm	Sets the power spectral density of trace E to be displayed on trace F.	
G=C-F(LOG)	Sets trace G to TRACE C-F computation mode (LOG).	
G=F-C(LOG)	Sets trace G to TRACE F-C computation mode (LOG).	
G=C+F(LOG)	Sets trace G to TRACE C+F computation mode (LOG).	
G=E-F(LOG)	Sets trace G to TRACE E-F computation mode (LOG).	
G=F-E(LOG)	Sets trace G to TRACE F-E computation mode (LOG).	
G=E+F(LOG)	Sets trace G to TRACE E+F computation mode (LOG).	
G=C+F(LIN)	Sets trace G to TRACE C+F computation mode (LIN).	
G=C-F(LIN)	Sets trace G to TRACE C-F computation mode (LIN).	
G=F-C(LIN)	Sets trace G to TRACE F-C computation mode (LIN).	
G=E+F(LIN)	Sets trace G to TRACE E+F computation mode (LIN).	
G=E-F(LIN)	Sets trace G to TRACE E-F computation mode (LIN).	
G=F-E(LIN)	Sets trace G to TRACE F-E computation mode (LIN).	
G=NORM A	Sets the normalizd data of trace A to be displayed on trace G.	
G=NORM B	Sets the normalizd data of trace B to be displayed on trace G.	
G=NORM C	Sets the normalizd data of trace C to be displayed on trace G.	
G=CURVE FIT A **dB	Sets curve fit processed data from TRACE A to be displayed on trace G.	0 to 99 (1 step)
G=CURVE FIT B **dB	Sets curve fit processed data from TRACE B to be displayed on trace G.	0 to 99 (1 step)
G=CURVE FIT C **dB	Sets curve fit processed data from TRACE C to be displayed on trace G.	0 to 99 (1 step)
G=MARKER FIT **dB	Sets curve fit processed data from the placed delta marker to be displayed on trace G.	0 to 99 (1 step)
G=CURVE FIT PEAK A **dB	Sets peak fit processed data from TRACE A to be displayed on trace G.	0 to 99 (1 step)
G=CURVE FIT PEAK B **dB	Sets peak curve fit processed data from race B to be displayed on trace G.	0 to 99 (1 step)
G=CURVE FIT PEAK C **dB	Sets peak curve fit processed data from trace C to be displayed on trace G.	0 to 99 (1 step)
CVFIT OPERATION AREA ####	Sets the target range for calculation when creating curve fit processed data.	####: ALL/IN L1-L2/OUT L1-L2
CURVE FIT/CURVE FIT PEAK ALGO ####	Sets the fitting function when creating a fitting function.	####:GAUSS/LORENZ/ 3RD POLY/4TH POLY/5TH POLY
TRACE #->@	Copies data from TRACE of the variable # to TRACE of the variable @	#: A to G
TRACE # CLEAR	Clears trace # data.	#: A to G
ALL TRACE CLEAR	Clears all trace data.	

Program Command	Description	Parameter ranges and supported variables.
LABEL '---56 chars---'	Displays a label comment in the label area. If a semicolon (;) is added to the end, the comment (variable value) specified by the next LABEL command is displayed.	
LABEL @@@@@	Sets the contents of variable @@@@@ to the label area.	@@@@@: E, G, H, I, J, K, O, P, Q, R, S, T, U, V, X, Y, Z, WM, W1, W2, W(CH), W2-W1, LM, L1, L2, L2-L1, L(CH), SPWD, MEANWL, PKWL, PKLVL, MODN, GONO, SMSR, SMSR2, WDMCHN, WDMWL(CH), WDMLVL(CH), WDMSNR(CH), NFCHN, NFWL(CH), NFLVLI(CH), NFLVLO(CH), NFASELV(CH), NFGAIN(CH), NFNF(CH), MKPWR, DOMWL, XCOL, YCOL, ZCOL, M, N, CH, A$, B$, C$, D$, FILE$, TIME$
LABEL @@@@@;	Sets the contents of variable @@@@@ to the label display. The comment (variable value) specified by the next LABEL command is displayed.	@@@@@: E, G, H, I, J, K, O, P, Q, R, S, T, U, V, X, Y, Z, WM, W1, W2, W(CH), W2-W1, LM, L1, L2, L2-L1, L(CH), SPWD, MEANWL, PKWL, PKLVL, MODN, GONO, SMSR, SMSR2, WDMCHN, WDMWL(CH), WDMLVL(CH), WDMSNR(CH), NFCHN, NFWL(CH), NFLVLI(CH), NFLVLO(CH), NFASELV(CH), NFGAIN(CH), NFNF(CH), MKPWR, DOMWL, XCOL, YCOL, ZCOL, M, N, CH, A$, B$, C$, D$, FILE$, TIME$
LABEL CLEAR	Clears the LABEL command in the label area.	
NOISE MASK ***dB	Displays waveform data with the data at or below the set level masked.	OFF (–999), –100 to 0 (1 step)
MASK LINE VERTICAL	Sets the mask value in the noise mask function or lower to zero.	
MASK LINE HORIZONTAL	Sets the mask value in the noise mask function or lower to the mask value.	

ZOOM

Program Command	Description	Parameter ranges and supported variables.
ZOOM CENTER WL ****.***nm	Sets the display scale's center wavelength.	350.000 to 1750.000 (0.001 step)
ZOOM CENTER @@@@@	Sets the value of variable @@@@@ to the display scale center wavelength.	@@@@@: E, G, H, I, J, K, O, P, Q, R,S, T, U, V, X, Y, Z, WM, W1, W2, W(CH), PKWL, DOMWL, MEANWL, WDMWL(CH), NFWL(CH)
ZOOM CENTER FREQ ***.****THz	Sets the display scale's center frequency.	171.0000 to 857.0000 THz (0.0001 step)
ZOOM CENTER FREQ @@@@@	Sets the value of variable @@@@@ to the display scale center frequency.	@@@@@: E, G, H, I, J, K, O, P, Q, R,S, T, U, V, X, Y, Z, PKWL, DOMWL, MEANWL, WDMWL(CH), NFWL(CH)
ZOOM CENTER WNUM****.***cm^{-1}	Sets the display scale's center wavenumber	5714.000 to 28571.000 cm^{-1} (0.001 step)
ZOOM CENTER WNUM@@@@@	Sets the value of variable @@@@@ to the display scale center wavenumber.	@@@@@: E, G, H, I, J, K, O, P, Q, R,S, T, U, V, X, Y, Z, PKWL, DOMWL, MEANWL, WDMWL(CH), NFWL(CH)
ZOOM SPAN WL ****.*nm	Sets the display scale's span.	0.5 to 1400.0 nm (0.1 step)
ZOOM SPAN WL @@@@@	Sets the value of variable @@@@@ to the display scale span	@@@@@: E, G, H, I, J, K, O, P, Q, R,S, T, U, V, X, Y, Z, W2-W1, SPWD
ZOOM SPAN FREQ ***.**THz	Sets the display scale's span.	0.05 to 686.00 THz (0.01 step)
ZOOM SPAN FREQ @@@@@	Sets the value of variable @@@@@ to the display scale span.	@@@@@: E, G, H, I, J, K, O, P, Q, R,S, T, U, V, X, Y, Z, W2-W1, SPWD
ZOOM SPAN WNUM ****.*cm^{-1}	Sets the display scale's span.	0.1 to 22858.0 cm^{-1} (0.1 step)
ZOOM SPAN WNUM @@@@@	Sets the value of variable @@@@@ to the display scale span.	@@@@@: E, G, H, I, J, K, O, P, Q, R,S, T, U, V, X, Y, Z, W2-W1, SPWD
ZOOM START WL ****.***nm	Sets the starting wavelength of the display scale.	1.000 to 1749.950 nm (0.001 step)
ZOOM START FREQ ***.****THz	Sets the starting frequency of the display scale.	10.0000 to 856.9950 THz (0.0001 step)
ZOOM START WNUM ***.****cm^{-1}	Sets the starting wavenumber of the display scale.	1000.000 to 28570.950 cm^{-1} (0.001 step)
ZOOM STOP WL ****.***nm	Sets the ending wavelength of the display scale.	350.050 to 2450.000 nm (0.001 step)
ZOOM STOP FREQ ***.****THz	Sets the ending frequency of the display scale.	171.0050 to 999.9000 (0.0001 step)
ZOOM STOP WNUM ***.****cm^{-1}	Sets the ending wavenumber of the display scale.	5714.050 to 40000.000 cm^{-1} (0.001 step)
PEAK->ZOOM CENTER	Sets the peak wavelength of the waveform on the active trace. Sets the wavelength to the display scale's center wavelength.	
OVERVIEW DISPLAY OFF	Sets OVERVIEW display during ZOOM to OFF.	
OVERVIEW DISPLAY LEFT	Sets OVERVIEW display during ZOOM to the left side of the waveform screen.	
OVERVIEW DISPLAY RIGHT	Sets OVERVIEW display during ZOOM to the right side of the waveform screen.	
ZOOM INITIALIZE	Resets the display scale to the initial state.	

MARKER

Program Command	Description	Parameter ranges and supported variables.
MARKER ****.***nm	Sets the marker to the specified wavelength position on the active trace (according to the wavelength value).	350.000 to 1750.000 (0.001 step)
MARKER ***.****THz	Sets the marker to the specified wavelength position on the active trace (according to the frequency value).	171.0000 to 857.0000 (0.0001 step)
MARKER ****.***cm^{-1}	Sets the marker to the specified wavelength position on the active trace (according to the wavenumber value).	5714.000 to 28571.000 cm^{-1} (0.001 step)
MARKER @@@@@	Sets a marker to the wavelength position of variable @@@@@.	@@@@@: E, G, H, I, J, K, O, P, Q, R,S, T, U, V, X, Y, Z, WM, W1, W2, W(CH), MEANWL, PKWL, DOMWL, WDMWL(CH), NFWL(CH)
SET MARKER ****	Sets fixed marker **** to the moving marker position.	1 to 1024 (1 step)
SET MARKER @	Sets the fixed marker of variable @ to the moving marker position.	@: E, G, H, I, J, K, O, P, Q, R,S, T, U, V, X, Y, Z
CLEAR MARKER ****	Clears fixed marker ****.	1 to 1024 (1 step)
CLEAR MARKER @	Clears the fixed marker of variable @.	@: E, G, H, I, J, K, O, P, Q, R,S, T, U, V, X, Y, Z, N, M
MARKER TYPE #####	Sets the type of marker.	####: NORMAL, DENSITY, INTEGRAL
MARKER INTEGRAL RANGE **.*GHz	Sets the integration range of the marker.	***.* : 1.0 to 999.9 (0.1 step)
MARKER BANDWIDTH **.*nm	Sets the normalization bandwidth of the power spectral density marker	**.* : 0.1 to 10.0 (0.1 step)
MARKER->CENTER	Sets the wavelength value of a marker to the measurement center wavelength.	
MARKER-> ZOOM CENTER	Sets the wavelength value of a marker to the display scale's center wavelength.	
MARKER->REF LEVEL	Sets the marker level value to thereference level.	
ALL MARKER CLEAR	Sets the advanced marker to the specified wavelength position (according to the wavelength value).	
LINE MARKER1 ****.***nm	Sets line marker 1 to a specified wavelength position (according to a wavelength value).	350.000 to 1750.000 (0.001 step)
LINE MARKER1 ***.****THz	Sets line marker 1 to a specified frequency position (according to a frequency value).	171.0000 to 857.0000 (0.0001 step)
LINE MARKER1 ****.***cm^{-1}	Sets line marker 1 to a specified wavenumber position (according to a wavenumber value).	5714.000 to 28571.000 cm-1 (0.001 step)
LINE MARKER1 @@@@@	Sets line marker 1 to the wavelength position of vaiable @@@@@.	@@@@@:E, G, H, I, J, K, O, P, Q, R,S, T, U, V, X, Y, Z, WM, W1, W2, W(CH), MEANWL, PKWL, DOMWL, WDMWL(CH), NFWL(CH)
LINE MARKER2 ****.***nm	Sets line marker 2 to a specified wavelength position (according to a wavelength value).	350.000 to 1750.000 (0.001 step)
LINE MARKER2 ***.****THz	Sets line marker 2 to a specified frequency position (according to a frequency value).	171.0000 to 857.0000 (0.0001 step)
LINE MARKER2 ****.***cm^{-1}	Sets line marker 2 to a specified wavenumber position (according to a wavenumber value).	5714.000 to 28571.000 cm^{-1} (0.001 step)

Program Command	Description	Parameter ranges and supported variables.
LINE MARKER2 @@@@@	Sets line marker 2 to the wavelength position of variable @@@@@	@@@@@: E, G, H, I, J, K, O, P, Q, R,S, T, U, V, X, Y, Z, WM, W1, W2, W(CH), MEANWL, PKWL, DOMWL, WDMWL(CH), NFWL(CH)
LINE MARKER3 ***.**dB	Sets line marker 3 to a specified level.	–139.90 to 159.90 (0.01 step)
LINE MARKER3 ***.**dBm	Sets line marker 3 to a specified level.	–210.00 to 50.00 (0.01 step)
LINE MARKER3 *.***##	Sets line marker 3 to a specified level.	1.00pW to 1000mW (1.00 to 9.99[pW, nW, mW, mW]: 0.01 step 10.0 to 99.9(100) [pW, nW, mW, (mW)]: 0.1 step 100 to 999 [pW, nW, mW, mW]: 1 step) ## is , pW, nW, mW, mW (Select one of the above)
LINE MARKER3 **.***	Sets line marker 3 to a specified level.	0.00 to 2500.00 (0.01 step)
LINE MARKER3 @@@@@	Sets line marker 3 to the level position of variable @@@@@	@@@@@: E, G, H, I, J, K, O, P, Q, R,S, T, U, V, X, Y, Z, LM, L1, L2, L(CH), PKLVL, WDMLVL(CH), NFLVI(CH), NFLVO(CH), NFASELV(CH), MKPWR
LINE MARKER4 ****.***dB	Sets line marker 4 to a specified level.	–139.90 to 159.90 (0.01 step)
LINE MARKER4 ****.***dBm	Sets line marker 4 to a specified level.	–210.00 to 50.00 (0.01 step)
LINE MARKER4 *.***##	Sets line marker 4 to a specified level.	1.00pW to 1000mW (1.00 to 9.99[pW, nW, µW, mW]: 0.01 step 10.0 to 99.9(100) [pW, nW, µW, (mW)]: 0.1 step 100 to 999 [pW, nW, µW, mW]: 1 step) ## is , pW, nW, µW, mW(Select one of the above)
LINE MARKER4 **.***	Sets line marker 4 to a specified level.	0.00 to 2500.00 (0.01 step)
LINE MARKER4 @@@@@	Sets line marker 4 to the level position of variable @@@@@	@@@@@: E, G, H, I, J, K, O, P, Q, R,S, T, U, V, X, Y, Z, LM, L1, L2, L(CH), PKLVL, WDMLVL(CH), NFLVLI(CH), NFLVLO(CH), NFASELV(CH), MKPWR
MARKER L1-L2->SPAN	Sets the range surrounded by line markers 1 and 2 to the measuring span.	
MARKER L1-L2->ZOOM SPAN	Sets the range surrounded by line markers 1 and 2 to the display scale span.	
LINE MARKER CLEAR	Clears line markers on the screen.	
MARKER OFFSET LIST	Displays the difference from the moving marker.	
MARKER SPACING LIST	Displays a difference to a neighboring marker.	
MARKER AUTO UPDATE ###	Makes the level position of a fixed marker follow the active trace waveform.	###: ON/OFF
MARKER UNIT nm	Sets a wavelength marker value to the wavelength display.	
MARKER UNIT THz	Sets a wavelength marker value to the frequency display.	
MARKER UNIT cm^{-1}	Sets a wavelength marker value to the wavenumber display.	
SEARCH/ANA L1-L2 ###	Selects ON/OFF for the analysis function in the range surrounded by line markers 1 and 2	###: ON/OFF
SEARCH/ANA ZOOM AREA ###	Selects ON/OFF for the analysis function of the display scale range.	###: ON/OFF

PEAK SERCH

Program Command	Description	Parameter ranges and supported variables.
PEAK SEARCH	Performs a peak search on the active trace waveform.	
BOTTOM SEARCH	Performs a bottom search on the active trace waveform.	
NEXT SRCH	Searches for the next peak/bottom after the peak/bottom level of the active trace waveform.	
NEXT SRCH RIGHT	Searches for the peak/bottom to the right of the peak/bottom marker of the active trace waveform.	
NEXT SRCH LEFT	Searches for the peak/bottom to the left of the peak/bottom marker of the active trace waveform.	
AUTO SEARCH ###	Selects ON/OFF of the peak/bottom search function conducted each sweep.	###: ON/OFF
MODE DIFF **.**dB	Sets the level difference of the mode judgment criteria used for peak search or waveform analysis.	0.01–50.00 (0.01 step)
SEARCH MODE ######	Sets the search mode.	######: SINGLE/MULTI
MULTI SEARCH THRESH **.**dB	Sets the multi search threshold.	0.01: 99.99 (0.01 step)
MULTI SEARCH SORT BY #####	Sets the multi search detection list sort order.	#####: WL/LEVEL

ANALYSIS

Program Command	Description	Parameter ranges and supported variables.
SPEC WD THRESH **.**dB	Performs a THRESH-based spectrum width search according to the specified threshold value.	0.01 to 50.00 (0.01 step)
PARAM THRESH K **.**	Sets the magnification for the THRESH based spectrum width search.	1.00 to 10.00 (0.01 step)
PARAM THRESH MODE FIT ###	Turns ON/OFF the function that sets the marker to the peak of the mode when performing a THRESH-based spectrum width search.	###: ON/OFF
SPEC WD ENV **.**dB	Performs an envelope-based spectrum width search using the specified threshold value.	0.01 to 50.00 (0.01 step)
PARAM ENV TH2 **.**dB	Sets the cutoff value for the envelopebased spectrum width search.	0.01 to 50.00 (0.01 step)
PARAM ENV K **.**dB	Sets the cutoff value for the envelopebased using the THRESH method.	1.00 to 10.00 (0.01 step)
SPEC WD RMS **.**dB	Performs an RMS-based spectrum width search according to a specified threshold.	0.01 to 50.00 (0.01 step)
PARAM RMS K **.**	Sets the magnification for an RMSbased spectrum width search.	1.00 to 10.00 (0.01 step)
SPEC WD PEAK RMS **.**dB	Performs an RMS-based spectrum width search according to a specified threshold value.	0.01 to 50.00 (0.01 step)
PARAM PEAK RMS K **.**	Sets the magnification for a PEAKRMS-based spectrum width search.	1.00 to 10.00 (0.01 step)
SPEC WD NOTCH **.**dB	Measures the NOTCH width using a specified threshold value.	0.01 to 50.00 (0.01 step)

Program Command	Description	Parameter ranges and supported variables.
PARAM NOTCH K **.**	Sets a magnification based on notch width measurement.	1.00 to 10.00 (0.01 step)
NOTCH FROM ######	Sets the reference for making notch width measurements.	#####: PEAK/BOTTOM
SMSR *	Sets the execution mode applied in SMR measurement.	1, 2, 3, 4
SMSR MASK ± **.**nm	Sets a mask range close to the peak during SMSR1 measurements.	0.00 to 99.99 (0.01 step)
POWER	Performs power analysis.	
POWER OFFSET ***.**dB	Sets a correction value in power measurements	–10.00 to 10.00 (0.01 step)
DFB-LD ANALYSIS	Performs analysis necessary for DFBLD.	
FP-LD ANALYSIS	Performs analysis necessary for FPLD.	
LED ANALYSIS	Performs analysis necessary for LED.	
ITLA ANALYSIS	Performs analysis necessary for ITLA.	
WDM ANALYSIS	Performs analysis necessary for WMD.	
WDM THRESH **.**dB	Sets a threshold value for WDM analysis	0.1 to 99.9 (0.1 step)
WDM MODE DIFF **.**dB	Sets the minimum peak/bottom difference for channel detection during WDM analysis.	0.01 to 50.00 (0.01 step)
WDM DISPLAY MASK OFF	Cancels level threshold value settingwhen masking display channels.	
WDM DISPLAY MASK ****.**dB	Sets the level threshold value when masking display channels.	–100.00 to 0.00 (0.01 step)
WDM NOISE ALGO AUTO FIX	Sets noise level measuring algorithm to AUTO.	
WDM NOISE ALGO MANUAL FIX	Sets noise level measuring algorithmto MANUAL FIX.	
WDM NOISE ALGO AUTO CTR	Sets noise level measuring algorithmto AUTO CTR.	
WDM NOISE ALGO MANUAL CTR	Sets noise level measuring algorithmto MANUAL CTR.	
WDM NOISE ALGO PIT	Sets noise level measuring algorithmto PIT.	
WDM NOISE AREA **.**nm	Sets an area used for noise level analysis in a range centered on channel wavelength.	0.01 to 10.00 (0.01 step)
WDM NOISE AREA @	Sets an area used for noise level analysis in a range of variable @ centered on channel channel wavelength.	@: E, G, H, I, J, K, O, P, Q, R,S, T, U, V, X, Y, Z
WDM MASK AREA **.**nm	Sets the signal light spectrum range to mask as centered on channel wavelength.	0.01 to 10.00 (0.01 step)
WDM MASK AREA @	Sets the signal light spectrum range to mask as centered on channel wavelength, to the range of variable @	@: E, G, H, I, J, K, O, P, Q, R,S, T, U, V, X, Y, Z
WDM FITTING ALGO LINEAR	Sets the fitting algorithm for finding noise level to linear interpolation mode.	
WDM FITTING ALGO GAUSS	Sets the fitting algorithm for finding noise level to normal distribution curve mode.	
WDM FITTING ALGO LORENZ	Sets the fitting algorithm for finding noise level to Lorenz curve mode.	
WDM FITTING ALGO 3RD POLY	Sets the fitting algorithm for finding noise level in 3rd polynomial mode.	
WDM FITTING ALGO 4TH POLY	Sets the fitting algorithm for finding noise level in 4th polynomial mode.	
WDM FITTING ALGO 5TH POLY	Sets the fitting algorithm for finding noise level in 5th polynomial mode.	

Program Command	Description	Parameter ranges and supported variables.
WDM NOISE BANDWIDTH *.**nm	Sets bandwidth applied in measuring noise.	0.01 to 1.00 (0.01 step)
WDM DUAL TRACE ###	Makes setting so that both TRACEs A and B are used in analyzing WDM.	###: ON/OFF
WDM DISPLAY ABSOLUTE	Sets the display of WDM analysis results to absolute value display.	
WDM DISPLAY RELATIVE	Sets the display of WDM analysis results to relative value display.	
WDM DISPLAY DRIFT MEAS	Sets the display of WDM analysis results to drift value display (drift display using past measurement wavelength as a reference)	
WDM DISPLAY DRIFT GRID	Sets the display of WDM analysis results to drift value display (using grid wavelength as a reference)	
WDM CH RELATION #######	Sets the display format of an interchannel level absolute value when WDM analysis display is in absolute value display.	####### OFFSET/SPACING
WDM CHANNEL HIGHEST	Sets the reference channel when the CH RELATION is OFFSET to the channel with the highest level.	
WDM CHANNEL NO.****	Sets the reference channel when the CH RELATION is OFFSET.	1 to 1024 (1 step)
WDM MAX/MIN RESET	Resets MAX/MIN data during DRIFT (MEAS, GRID) measurement.	
WDM OUTPUT SLOPE ###	Displays the least square approximation line of a channel peak.	###: ON/OFF
WDM POINT DISPLAY ###	Displays the range of data used in fitting on the waveform screen.	###: ON/OFF
WDM OSNR DISPLAY ###	Displays the noise and OSNR, which are the results of WDM analysis.	###: ON/OFF
WDM SIGNAL POWER ########	Sets the signal optical power calculation method.	########: PEAK / INTEGRAL
WDM INTEGRAL RANGE ***GHz	Sets the signal optical power integral range.	1.0 to 999.9 (0.1step)
WDM INTEGRAL RANGE **.**dB	Sets the threshold level of the signal power integral range.	0.01 to 99.99 (0.01 step)
EDFA NF ANALYSIS	Performs analysis necessary for EDFA-NF measurements.	
EDFA NF THRESH **.**dB	Sets an EDFA-NF analysis threshold.	0.1 to 99.9 (0.1 step)
EDFA NF MODE DIFF **.**dB	Sets the minimum peak/bottom difference for channel detection during EDFA-NF analysis.	0.01 to 50.00 (0.01 step)
EDFA NF OFFSET(IN) ***.**dB	Sets a signal light offset value used for NF and Gain calculation.	–99.99 to 99.99 (0.01 step)
EDFA NF OFFSET(IN) @	Sets the offset value of the signal used for NF and Gain calculation to the variable @.	@:E, G, H, I, J, K, O, P, Q, R,S, T, U, V, X, Y, Z
EDFA NF OFFSET(OUT) ***.**dB	Sets an output light offset value used for NF and Gain calculation.	–99.99 to 99.99 (0.01 step)
EDFA NF OFFSET(OUT) @	Sets an output light offset value used for NF and Gain calculation to the variable @.	@: E, G, H, I, J, K, O, P, Q, R,S, T, U, V, X, Y, Z
EDFA NF ASE ALGO AUTO FIX	Sets the ASE level measuring algorithm to ATUO FIX.	
EDFA NF ASE ALGO MANUAL FIX	Sets the ASE level measuring algorithm to MANUAL FIX.	

Program Command	Description	Parameter ranges and supported variables.
EDFA NF ASE ALGO AUTO CTR	Sets the ASE level measuring algorithm to AUTO CTR.	
EDFA NF ASE ALGO MANUAL CTR	Sets the ASE level measuring algorithm to MANUAL CTR.	
EDFA NF ASE AREA **.**nm	Sets an area used for ASE level analysis in a range centered on channel wavelength.	0.01 to 10.00 (0.01 step)
EDFA NF ASE AREA @	Sets an area used for ASE level analysis in a range centered on variable @@@@@.	@@@@: E, G, H, I, J, K, O, P, Q, R,S, T, U, V, X, Y, Z
EDFA NF MASK AREA **.**nm	Sets the signal light spectrum range to mask as centered on channel wavelength.	0.01 to 10.00 (0.01 step)
EDFA NF MASK AREA @	Sets the signal light spectrum range to mask as centered on variable @@@@@.	@@@@: E, G, H, I, J, K, O, P, Q, R,S, T, U, V, X, Y, Z
EDFA NF FITTING ALGO LINEAR	Sets the fitting algorithm for finding ASE level to linear interpolation mode.	
EDFA NF FITTING ALGO GAUSS	Sets the fitting algorithm for finding ASE level to normal distribution curve mode.	
EDFA NF FITTING ALGO LORENZ	Sets the fitting algorithm for finding ASE level to Lorenz curve mode.	
EDFA NF FITTING ALGO 3RD POLY	Sets the fitting algorithm for finding ASE level in 3rd polynomial mode.	
EDFA NF FITTING ALGO 4TH POLY	Sets the fitting algorithm for finding ASE level in 4th polynomial mode.	
EDFA NF FITTING ALGO 5TH POLY	Sets the fitting algorithm for finding ASE level in 5th polynomial mode.	
EDFA NF POINT DISPLAY ###	Displays the range of data used in fitting on the waveform screen.	###: ON/OFF
EDFA NF RES BW MEASURED	For the resolution, use the value determined from the waveform using THRESH 3dB analysis.	
EDFA NF RES BW CAL DATA	For the resolution, use the actual resolution value stored in the instrument.	
EDFA NF SHOT NOISE ###	Set whether to include/not include Shot Noise in the NF computation.	
EDFA NF SIGNAL POWER ########	Sets the signal optical power calculation method.	
EDFA NF INTEGRAL RANGE ***.*GHz	Sets the signal optical power integral range.	
FILTER(PEAK) ANALYSIS	Performs optical filter (PEAK) analysis.	
FILTER(BOTTOM) ANALYSIS	Performs optical filter (BOTTOM) analysis.	
WDM FILTER(PEAK) ANALYSIS	Performs multi-channel type optical filter (PEAK) analysis.	
WDM FILTER(BOTTOM) ANALYSIS	Performs multi-channel type optical filter (BOTTOM) analysis.	
COLOR ANALYSIS	Performs a color analysis.	
WDM SMSR ANALYSIS	Performs multi-channel type SMSR analysis.	
SWITCH DISPLAY TO TRACE&TABLE	Displays both waveforms and tables in the display of analysis results.	
SWITCH DISPLAY TO TABLE	Displays only tables in the display of analysis results.	
SWITCH DISPLAY TO TRACE	Displays only traces in the display of analysis results.	

Program Command	Description	Parameter ranges and supported variables.
SWITCH DISPLAY TO GRAPH&TABLE	Displays both graphs and tables in the display of analysis results.	
SWITCH DISPLAY TO GRAPH	Displays only graphs in the displayof analysis results.	
AUTO ANALYSIS ###	Selects ON/OFF of the waveform analysis function activated each time a sweep is made.	
SYSTEM GRID 200GHz	Sets system grid to a 200 GHz spacing grid table.	
SYSTEM GRID 100GHz	Sets system grid to a 100 GHz spacing grid table.	
SYSTEM GRID 50GHz	Sets system grid to a 50 GHz spacing grid table.	
SYSTEM GRID 25GHz	Sets system grid to a 25 GHz spacing grid table.	
SYSTEM GRID 12.5GHz	Sets system grid to a 12.5 GHz spacing grid table.	
CUSTOM GRID START WL ****.****nm	Inputs the user grid table start wavelength.	350.0000 to 1750.000 (0.0001 step)
CUSTOM GRID START FREQ ***.****THz	Inputs the user grid table start frequency.	171.0000 to 857.0000 (0.0001 step)
CUSTOM GRID STOP WL ****.****nm	Inputs the user grid table stop wavelength.	350.0000 to 1750.0000 (0.0001 step)
CUSTOM GRID STOP FREQ ***.****THz	Inputs the user grid table stop frequency.	171.0000 to 857.0000 (0.0001 step)
CUSTOM GRID SPACING ***.*GHz	Inputs the user grid table gridspacing.	0.100 to 999.999 (0.1 step)
GRID REFERENCE FREQ ***.*****THz	Inputs the reference frequency of the grid table.	171.0000 to 857.0000 (0.0001 step)

FILE

Program Command	Description	Parameter ranges and supported variables.
WRITE TRACE # INT: '########.***'	Assign a file name to specified TRACE data and save itto internal memory.	#: A, B, C, D, E, F, G '########.***': file name
WRITE TRACE # EXT: '########.***'	Assign a file name to specified TRACE data and save it in external memory.	#: A, B, C, D, E, F, G '########.***': file name
WRITE TRACE # INT	Saves specified TRACE data in internal memory. File names are assigned automatically.	#: A, B, C, D, E, F, G
WRITE TRACE # EXT	Saves specified TRACE data inexternal memory. File names are assigned automatically.	#: A, B, C, D, E, F, G
WRITE TRACE # INT @@	Saves specified TRACE data in internal memory under the file namespecified in the variable @@.	#: A, B, C, D, E, F, G @@: A$, B$, C$, D$
WRITE TRACE # EXT @@	Saves specified TRACE data in external memory under the file namespecified in the variable @@.	#: A, B, C, D, E, F, G @@: A$, B$, C$, D$
TRACE WRITE:BINARY	Sets the data storage format to BINARY.	
TRACE WRITE:CSV	Sets the data storage format to CSV.	
WRITE ALL TRACE INT: '########.***'	Specify a file name for all TRACE data and save to internal memory.	'########.***': file name
WRITE ALL TRACE EXT: '########.***'	Specify a file name for all TRACE data and save to external memory.	'########.***': file name
WRITE ALL TRACE INT	Save all TRACE data to internal memory. A file name is automatically assigned.	

Program Command	Description	Parameter ranges and supported variables.
WRITE ALL TRACE EXT	Save all TRACE data to external memory. A file name is automatically assigned.	
WRITE ALL TRACE INT @@	Save all TRACE data under file names specified by the @@ variable to internal memory.	@@: A$, B$, C$, D$
WRITE ALL TRACE EXT @@	Save all TRACE data under file names specified by the @@ variable to external memory.	@@: A$, B$, C$, D$
WRITE GRAPH INT: '########.***'	Specifies a file name and saves graphic data in internal memory.	'########.***': file name
WRITE GRAPH EXT: '########.***'	Specifies a file name and saves graphic data in external memory.	'########.***': file name
WRITE GRAPH INT	Saves graphic data in internal memoryFile names are assigned automatically.	
WRITE GRAPH EXT	Saves graphic data in external memory. File names are assigned automatically.	
WRITE GRAPH INT @@	Saves graphic data under the file name specified by the variable @@ in internal memory.	@@: A$, B$, C$, D$
WRITE GRAPH EXT @@	Saves graphic data under the file name specified by the variable @@ in external memory.	@@: A$, B$, C$, D$
GRAPH COLOR MODE:B&W	Sets the graphic color mode to black & white.	
GRAPH COLOR MODE:COLOR	Sets the graphic color mode to screencolor mode.	
GRAPH COLOR MODE:PRESET COLOR	Sets the graphic color mode to PRESET COLOR (waveform in color, background in black & white).	
GRAPH TYPE:BMP	Sets the graphic file type to BMP.	
WRITE SETTING INT: '########.STG'	Specifies a file name and saves setting data to internal memory.	'########.STG': file name
WRITE SETTING EXT: '########.STG'	Specifies a file name and saves setting data to external memory.	'########.STG': file name
WRITE SETTING INT	Saves setting data to internal memory. File names are assigned automatically.	
WRITE SETTING EXT	Saves setting data to external memory. File names are assigned automatically.	
WRITE SETTING INT @@	Saves setting data under the file name specified in the variable @@ to internal memory.	@@: A$, B$, C$, D$
WRITE SETTING EXT @@	Saves setting data under the file name specified in the variable @@ to external memory.	@@: A$, B$, C$, D$
DATA:ADD WRITE	Writes an added data file.	
DATA:OVER WRITE	Overwrites a data file.	
DATA WRITE:CSV	Sets the data storage format to CSV.	
WRITE DATA INT:'#######.CSV'	Specifies a file name and saves data to internal memory.	'########.***': file name
WRITE DATA EXT:'#######.CSV'	Specifies a file name and saves data to external memory	'########.***': file name
WRITE DATA INT	Specifies a file name and saves datato internal memory. File names are assigned automatically.	
WRITE DATA EXT	Specifies a file name and saves data to external memory. File names are assigned automatically.	

Program Command	Description	Parameter ranges and supported variables.
WRITE DATA INT @@	Specifies a file name and saves data under the file name specified by the variable @@ in internal memory.	@@: A$, B$, C$, D$
WRITE DATA EXT @@	Specifies a file name and saves data under the file name specified by the variable @@ in external memory.	@@: A$, B$, C$, D$
DATA DATE&TIME ###	Selects ON/OFF of date and time output.	###: ON/OFF
DATA LABEL ###	Selects ON/OFF of label output.	###: ON/OFF
DATA DATA AREA ###	Selects ON/OFF of data area output.	###: ON/OFF
DATA CONDITION ###	Selects ON/OFF of measuring conditions output.	###: ON/OFF
DATA TRACE DATA ###	Selects ON/OFF of waveform data output.	###: ON/OFF
DATA OUTPUT WINDOW ###	Selects ON/OFF of contents output of the OUTPUT WINDOW PROGRAM function.	###: ON/OFF
WRITE LOGGING INT:#######.***	Specifies a file name and saves the logging data in internal memory.	'########.***': file name
WRITE LOGGING EXT:#######.***	Specifies a file name and saves the logging data in external memory.	'########.***': file name
WRITE LOGGING INT	Automatically assignes a file name and saves the logging data in internal memory.	
WRITE LOGGING EXT	Automatically assignes a file name and saves the logging data in external memory.	
WRITE LOGGING INT @@	Save logging data under file names specified by the @@ variable to internal memory	@@: A$: B$: C$: D$
WRITE LOGGING EXT @@	Save logging data under file names specified by the @@ variable to external memory.	@@: A$: B$: C$: D$
LOGGING SAVE CSV ###	Sets whether data logging results will be saved to a file in CSV format.	###: ON/OFF
LOGGING SAVE TRACE ###	Sets whether temporary saved waveform files will be saved when data logging results is saved.	###: ON/OFF
READ TRACE # INT: '########.***'	Assigns a file name to specified TRACE data and reads it frominternal memory.	'########.***': file name #: A, B, C, D, E, F, G
READ TRACE # EXT: '########.***'	Assigns a file name to specified TRACE data and reads it from external memory.	'########.***': file name #: A, B, C, D, E, F, G
READ TRACE # INT @@	Reads TRACE data in the file name specified by the variable @@ from internal memory.	#: A, B, C, D, E, F, G @@:A$, B$, C$, D$
READ TRACE # EXT @@	Reads TRACE data in the file name specified by the variable @@ from external memory.	#: A, B, C, D, E, F, G @@:A$, B$, C$, D$
READ ALL TRACE INT: '########.$$$'	Specify a file name for all TRACE data and reads from internal memory.	'########.$$$': file name
READ ALL TRACE EXT: '########.$$$'	Specify a file name for all TRACE data and reads from external memory.	'########.$$$': file name
READ ALL TRACE INT @@	Reads all TRACE data under file names specified by the @@ variable from internal memory.	@@: A$, B$, C$, D$
READ ALL TRACE EXT @@	Save all TRACE data under file names specified by the @@ variable from external memory.	@@: A$, B$, C$, D$
READ SETTING INT: '########.$$$'	Specifies a file name and reads setting data from internal memory.	'########.$$$': file name

Program Command	Description	Parameter ranges and supported variables.
READ SETTING EXT: '########.$$$'	Specifies a file name and saves setting data from external memory.	'########.$$$': file name
READ SETTING INT @@	Reads setting data of the file name specified by the variable @@ from internal memory.	@@: A$, B$, C$, D$
READ SETTING EXT @@	Reads setting data of the file name specified by the variable @@ from external memory.	@@: A$, B$, C$, D$
READ LOGGING INT :'########.$$$'	Specifies a file name and reads data from internal memory.	'########.$$$': file name
READ LOGGING EXT :'########.$$$'	Specifies a file name and reads data from external memory.	'########.$$$': file name
READ LOGGING INT @@	Reads data in the file name specified by the variable @@ from internal memory.	@@: A$, B$, C$, D$
READ LOGGING EXT @@	Reads data in the file name specified by the variable @@ from external memory.	@@: A$, B$, C$, D$
DELETE INT:'########.$$$'	Deletes files in internal memory.	'########.$$$': file name
DELETE EXT: '########.$$$'	Deletes files in external memory.	'########.$$$': file name
DELETE INT @@	Deletes files specified by the variable @@ from internal memory.	@@: A$, B$, C$, D$
DELETE EXT @@	Deletes files specified by the variable @@ from external memory.	@@: A$, B$, C$, D$
RENAME INT:## @@	Changes the names of files in internal memory specified by the variable ## to the file name specified by the variable @@.	##, @@: A$, B$, C$, D$
RENAME EXT:## @@	Changes the names of files in external memory specified by the variable ## to the file names specified by the variable @@.	##, @@: A$, B$, C$, D$
REMOVE USB STORAGE	Brings USB storage media online.	
AUTO FILE NAME NUMBER	Assign a file name with a serial number automatically.	
AUTO FILE NAME DATE	Assign a file name with a date automatically.	

SYSTEM

Program Command	Description	Parameter ranges and supported variables.
OPTICAL ALIGNMENT	Aligns the optical axis of a monochromator optical system.	
SELF WL CALIBRATION	Sets the light source to be wavelength calibrated for the internal light source.	
EXT WL CALIBRATION ****.***nm	Sets the light source to be wavelength calibrated for the external light source (laser type).	350.000 to 1750.000 (0.001 step)
EXT-GAS WL CALIBRATION****.***nm	Sets the light source to be wavelength calibrated for the external light source (gas cell type).	350.000 to 1750.000 (0.001 step)
EMIS LINE WL CALIBRATION ****.***nm	Sets the light source to be wavelength calibrated for the external light source (emission line light source type)	350.000 to 1750.000 (0.001 step)
WL SHIFT **.***nm	Sets the amount of wavelength shift.	–5.000 to 5,000 (0.001 step)
LEVEL SHIFT ***.***dB	Sets the amount of level shift.	–60.000 to 60,000 (0.001 step)
REMOTE INTERFACE: GP-IB	Sets the remote interface to GP-IB.	
REMOTE INTERFACE: ETHERNET	Sets the remote interface to Ethernet.	
REMOTE INTERFACE: NETWORK (VXI-11)	Sets the remote interface to Ethernet (VXI-11).	
SELECT COLOR *	Selects the display color of the screen.	0 to 1 (1 step)
UNCAL WARNING DISPLAY ###	Displays UNCAL and warning.	###: ON/OFF
BUZZER CLICK ###	Turns the key press click soundON/ OFF.	###: ON/OFF
BUZZER WARNING ###	Turns the warning/error buzzerON/OFF.	###: ON/OFF
LEVEL DISPLAY DIGIT *	Sets the number of displayed digits (decimal place) of the level data displayed under the marker area and ANALYSIS results.	1 to 3 (1 step)
AUTO OFFSET ###	Turns auto offset ON/OFF.	###: ON/OFF
AUTO OFFSET INTERVAL *** min	Sets the time interval for executing the auto offset function.	***: integer
MEASURE WAVELENGTH AIR	Sets the measurement wavelength to an air wavelength	
MEASURE WAVELENGTH VACUUM	Sets the measurement wavelength to a vacuum wavelength	
DISPLAY OFF	Turns the display OFF	
DISPLAY ON	Turns the display ON	

APP

Program Command	Description	Parameter ranges and supported variables.
DATA LOGGING START	Starts data logging	
DATA LOGGING ITEM########	Sets the data logging source	########: WDM, PEAK, MULTI-PEAK, DFBLD
DATA LOGGING MODE####	Sets the data logging mode (maximum channel mode or maximum logging mode)	####: MODE1 (MAX 1024 ch, 2001 times), MODE2 (MAX 256 ch, 10001 times)
DATA LOGGING INTERVAL #######	Sets the measurement interval of data logging	####: SWEEP TIME, 1 sec, 2 sec, 5 sec, 10 sec, 30 sec, 1 min, 2 min, 5 min , 10 min
DATA LOGGING TEST DURATION *******s	Sets the measurement duration of data logging (in seconds)	*******: 1 to 8639999 (1 step)
DATA LOGGING PEAK TH TYPE###	Sets how the threshold for detecting the data logging mode (peak or bottom) is specified	###: ABS, REL
DATA LOGGING PEAK TH (ABS)****.**dBm	Sets the threshold (absolute value) for detecting the data logging mode	****.**: 20.00 to -100.00
DATA LOGGING PEAK TH (REL)**.**dB	Sets the threshold (relative value) for detecting the data logging mode	**.**: 0.01 to 99.99
DATA LOGGING CH MATCHING TH ± *.**nm	Sets the threshold of the channel-matching wavelength λ for data logging	*.**: 0.01 to 1.00
DATA LOGGING MEMORY ########	Sets the temporary area for saving waveform files of data logging	###: INTERNAL, EXTERNAL
DATA LOGGING TRACE LOGGING ###	Sets whether waveforms will be logged during data logging	###: ON/OFF

List of Special Commands

General Commands

Program Command	Description	Parameter ranges and supported variables.
COPY ON	Produces a hard copy of the screen to file.	
GOTO ***	Makes a jump to line ***.	1 to 200 (1 step)
GOTO PROGRAM **	Makes a jump to program ** to run it from the first line. After completing running of program **, control returns to the original program. However, if there is an END command in program **, return to the jump source is not performed and the program ends. When a program is executed using this command, variables are not initialized.	
WAIT *****S	Makes a wait of **** seconds.	1 to 99999 (1 step)
PAUSE '---56 chars---'	Pauses execution of a program and causes a message window to appear. This window displays a message and an explanation of the CONTINUE key. Pressing the CONTINUE soft key closes the window and executes the program. If a program is started via GP-IB, no pause is made.	
VARIABLE CLEAR	Initializes all variables used in a program.	
END	Ends a program.	
INIT	Initializes all parameters, but does not clear variables.	
@=VAL(@$)	Converts the string in variable @ to a numerical value an substitutes the value into variable @.	@: E, G, H, I, J, K, O, P, Q, R,S, T, U, V, X, Y, Z @$: A$, B$, C$, D$
BEEP **	Buzzer sounds for ** x 100 msec.	1 to 10 (1 step)
REM '---56 chars---'	Defines a comment in the program list. This command is not processed, and the program proceeds to the next line.	

Loop Control

Program Command	Description	Parameter ranges and supported variables.
N=********	Substitutes a value into variable N.	1 to 99999999 (1 step)
N=@@@@@@	Copies the contents of variable @@@@@ to variable N.	@@@@@: MODN, WDMCHN, NFCHN, M
N-N-1;IF N<>0 GOTO ***	Subtracts "1" from variable N and, if the result is not "0," makes a jump toline ***.	1 to 200 (1 step)
M=********	Substitutes a value into variable M.	1 to 99999999 (1 step)
M=@@@@@@	Copies the contents of variable @@@@@ to variable M.	@@@@@: MODN, WDMCHN, NFCHN, N
M-M-1;IF M<>0 GOTO ***	Subtracts "1" from variable N and, if the result is not "0," makes a jump to line ***.	1 to 200 (1 step)

Variable Calculations

Program Command	Description	Parameter ranges and supported variables.
@ = ********** ###	Substitutes a value into variable @.For **********, a real number of 10 or fewer digits can be specified, including a sign and the decimal point.	@: E, G, H, I, J, K, O, P, Q, R,S, T, U, V, X, Y, Z, CH **********: -999999999 to 9999999999 (1 step) ###: nm, dB, dBm, pW, nW, mW, mW, W, THz, cm^{-1} ' '(without units)
@ = # + **********	Adds value ********** to variable # and substitutes the value into variable @.********* can be specified with a real number of 10 or fewerdigits, including a sign and the decimal point. By specifying a negative value, you can cause subtraction to be made from variable #.	@, #: E, G, H, I, J, K, O, P, Q, R,S, T, U, V, X, Y, Z, CH **********: –999999999 to 9999999999 (1 step)
@ = @@@@@	Copies the contents of variable @@@@@ to variable @.	@: E, G, H, I, J, K, O, P, Q, R,S, T, U, V, X, Y, Z, CH @@@@@: E, G, H, I, J, K, O, P, Q, R,S, T, U, V, X, Y, Z, WM, W1, W2, W2-W1, W(CH), LM, L1, L2, L2-L1, L(CH), SPWD, MEANWL, PKWL, PKLVL, MODN, SMSR, WDMCHN, WDMWL(CH), WDMLVL(CH), WDMSNR(CH), NFCHN, NFWL(CH), NFLVLI(CH), NFLVLO(CH), NFASELV(CH), NFGAIN(CH), NFNF(CH), MKPWR, DOMWL, XCOL, YCOL, ZCOL, M, N, CH
@ = @@@@@ + ##### @ = @@@@@ - ##### @ = @@@@@ * ##### @ = @@@@@ / #####	Performs addition, subtraction, multiplication, and/or division between variables.	####: E, G, H, I, J, K, O, P, Q, R,S, T, U, V, X, Y, Z, CH @@@@@: E, G, H, I, J, K, O, P, Q, R,S, T, U, V, X, Y, Z, WM, W1, W2, W2-W1, W(CH), LM, L1, L2, L2-L1, L(CH), SPWD, MEANWL, PKWL, PKLVL, MODN, SMSR, WDMCHN, WDMWL(CH), WDMLVL(CH), WDMSNR(CH), NFCHN, NFWL(CH), NFLVLI(CH), NFLVLO(CH), NFASELV(CH), NFGAIN(CH), NFNF(CH), MKPWR, DOMWL, XCOL, YCOL, ZCOL, M, N, CH,
@$ = @$	Copies string variable @$ to stringvariable @.	@$: A$, B$, C$, D$
@$ = MID(@$, @, @)	Substitutes @'s worth of characters in the string that is distant from the start of character variable @$ by the number of characters in the numerical variable @ into character variable @$.	@: E, G, H, I, J, K, O, P, Q, R,S, T, U, V, X, Y, Z @$: A$, B$, C$, D$
@$ = '---56 chars---'	Substitutes string to character variable @$. (56 chars max)	@$: A$, B$, C$, D$
@$ =@$+@$	Substitutes the character string obtained by concatenating character variable @$ and character variable @$ into character variable @$.	@$: A$: B$: C$: D$
@$ =STR(@)	Converts variable @ into a character string and substitutes it into character variable @$	@: E, G, H, I, J, K, O, P, Q, R,S, T, U, V, X, Y, Z @$: A$: B$: C$: D$
@$ =DATEINFO(###)	Substitutes the date and time into character variable @$.	@$: A$: B$: C$: D$ ####: DATE&TIME, DATE, TIME

Condition Judgement

Program Command	Description	Parameter ranges and supported variables.
Program Command	**Description**	**Parameter ranges and supported variables.**
IF F1 <= @@@@@ <= F2 GOTO ***	Value of variable @@@@@ is F1 or greaterIf less than F2, jumps to line ***	@@@@@: E, G, H, I, J, K, O, P, Q, R,S, T, U, V, X, Y, Z, WM, W1, W2, W2-W1, W(CH), LM, L1, L2, L2-L1, L(CH), SPWD, MEANWL, PKWL, PKLVL, MODN, SMSR, SMSR2, WDMCHN, WDMWL(CH), WDMLVL(CH), WDMSNR(CH), FNCHN, NFWL(CH), NFLVL(CH), NFLVI(CH), NFLVO(CH), NFASELV(CH), NFGAIN(CH), NFNF(CH), MKPWR, DOMWL, XCOL, YCOL, ZCOL, M, N, CH ***: 1 to 200 (1 step)
F1 = ********** ###	Substitutes a value into variable F1. For **********, a real number of 10 or fewer digits can be specified including a sign and the decimal point.	###: nm, dB, dBm, pW, nW, mW, mW, W, THz, cm^{-1}, ' '(without units) **********: –999999999 to 9999999999 (1 step)
F2 = **********	Substitutes a value into ### variable F2. For **********, a real number of 10 or fewer digits can be specified, including a sign and the decimal point.	###: nm, dB, dBm, pW, nW, mW, mW, W, THz, cm^{-1}, ' '(without units) **********: –999999999 to 9999999999 (1 step)
F1 = @@@@@	Copies the contents of variable @@@@@ to the variable F1.	@@@@@: E, G, H, I, J, K, O, P, Q, R,S, T, U, V, X, Y, Z, WM, W1, W2, W2-W1, W(CH), LM, L1, L2, L2-L1, L(CH), SPWD, MEANWL, PKWL, PKLVL, MODN, SMSR, SMSR2, WDMCHN, WDMWL(CH), WDMLVL(CH), WDMSNR(CH), NFCHN, NFWL(CH), NFLVLI(CH), NFLVLO(CH), NFASELV(CH), NFGAIN(CH), NFNF(CH), MKPWR, DOMWL, XCOL, YCOL, ZCOL, M, N, CH
F2 = @@@@@	Copies the contents of variable @@@@@ to the variable F2.	@@@@@: E, G, H, I, J, K, O, P, Q, R,S, T, U, V, X, Y, Z, WM, W1, W2, W2-W1, W(CH), LM, L1, L2, L2-L1, L(CH), SPWD, MEANWL, PKWL, PKLVL, MODN, SMSR, SMSR2, WDMCHN, WDMWL(CH), WDMLVL(CH), WDMSNR(CH), NFCHN, NFWL(CH), NFLVLI(CH), NFLVLO(CH), NFASELV(CH), NFGAIN(CH), NFNF(CH), MKPWR, DOMWL, XCOL, YCOL, ZCOL, M, N, CH,
@ = LEVEL (****.***nm)	Substitutes the level of the point of wavelength ****.*** nm on an active trace into variable @.	@: G, H, I, J, K, P, Q, R, S, X, Y, Z ****.***: 350.000 to 1750.000 (0.001 step)
@ = LEVEL (@@@@@)	Substitutes the level of the point of the wavelength @@@@@ (variable) on the active trace into variable @.	@: E, G, H, I, J, K, O, P, Q, R,S, T, U, V, X, Y, Z @@@@@: G, H, I, J, K, P, Q, R, S, X, Y, Z, WM, W1, W2, W(CH), MEANWL, PKWL, WDMLVL(CH), WDMWL(CH), NFWL(CH), DOMWL

Program Command	Description	Parameter ranges and supported variables.
IF @@@@@ < @@@@@ GOTO *** IF @@@@@ =< @@@@@ GOTO *** IF @@@@@ = @@@@@ GOTO *** IF @@@@@ <> @@@@@ GOTO ***	Compares the large and small relationship of two variables and if the conditions are met, makes a jump to line ***.	@@@@@: E, G, H, I, J, K, O, P, Q, R,S, T, U, V, X, Y, Z, WM, W1, W2, W2-W1, W(CH), LM, L1, L2, L2-L1, L(CH), SPWD, MEANWL, PKWL, PKLVL, MODN, SMSR, SMSR2, WDMCHN, WDMWL(CH), WDMLVL(CH), \ WDMSNR(CH), FNCHN, NFWL(CH), -NFLVLI(CH), NFLVLO(CH), NFASELV(CH), NFGAIN(CH), NFNF(CH), MKPWR, M, N, DOMWL, XCOL, YCOL, ZCOL, CH ***: 1 to 200 (1 step)

External Control

Program Command	Description	Parameter ranges and supported variables.
SEND LAN @$, '*****' ; '---56 chars---' *@$: computer name or IP address ****: Port number	Specifies the external instrument that is connected to the LAN connector and that is specified by the computer name, IP address, and port number as the listener, and sends the command and sends the command in single quotes (' '). Delimiter is value of SET DELIMITER	Port Numeber: 1024 to 65535 @ $: A$, B$, C$, D$
SEND LAN @$, '*****' ; '---56 chars---' ;@ *@$: computer name or IP address ****: Port number	Specifies the external instrument that is connected to the LAN connector and that is specified by the computer name, IP address,and port number as the listener, and sends the command and following the commnd in single quotes (' '), sends the value of the variable @. Delimiter is value of SET DELIMITER.	Port Number: 1024 to 65535 @: E, G, H, I, J, K, O, P, Q, R,S, T, U, V, X, Y, Z @ $: A$, B$, C$, D$
SEND LAN @$, '*****' , '--20 chars?';@ "?20 chars?' *@$: computer name or IP address ****: Port Number	Specifies the external instrument that isconnected to the LAN connector and that is specified by the computer name, IP address, and port number as the listener, and following the commnd in single quotes (' '), sends the value of variable @, as well as the command in single quotes. The delimiter is the setting value of SET DELIMITER.	Port Number: 1024 to 65535 @: E, G, H, I, J, K, O, P, Q, R,S, T, U, V, X, Y, Z @ $: A$, B$, C$, D$
SENDR LAN @$, ' ***** ' , '---56 chars---' ; @$ *@$: computer name or IP address ****: Port number	Sends a query command to the external instrument that is connected to the LAN connector and which is specified by the computer name, IP address, and port number stored in variable @$. Substitutes the message received from the external instrument into character variable @$. Up to 512 characters can be received. The delimiter is the setting value of SET DELIMITER.	Port Number: 1024 to 65535 @ $: A$, B$, C$, D$
SET DELIMITER ###	On the external instrument beingremotely controlled with the LAN port, sets the delimiter that is sent/received by the instrument.	###:CR, LF, CR+LF

Substitution of Measuring Conditions

Program Command	Description	Parameter ranges and supported variables.
@ = CENTER	Substitutes the current measurement center wavelength into variable @.	@: E, G, H, I, J, K, O, P, Q, R,S, T, U, V, X, Y, Z
@ = SPAN	Substitutes the current sweep width into variable @.	@: E, G, H, I, J, K, O, P, Q, R,S, T, U, V, X, Y, Z
@ = REF LEVEL	Substitutes the current reference level into variable @.	@: E, G, H, I, J, K, O, P, Q, R,S, T, U, V, X, Y, Z
@ = RESOLUTION	Substitutes the current measurement resolution into variable @.	@: E, G, H, I, J, K, O, P, Q, R,S, T, U, V, X, Y, Z
@ = SAMPLING POINT	Substitutes the current number of samples into variable @.	@: E, G, H, I, J, K, O, P, Q, R,S, T, U, V, X, Y, Z
@ = ZOOM CENTER	Substitutes the current display center wavelength into variable @.	@: E, G, H, I, J, K, O, P, Q, R,S, T, U, V, X, Y, Z
@ = ZOOM SPAN	Substitutes the current display width into variable @.	E, G, H, I, J, K, O, P, Q, R,S, T, U, V, X, Y, Z

User I/O

Program Command	Description	Parameter ranges and supported variables.
DATA INPUT '---56 chars---';@	Pauses program execution, and gets the value/string input into variable @by the user. The Input Window appears on screen displaying a character string in ' '. When variable @ is numerical it accepts numerical input and when it is a string variable it accepts string input.	@@@@@: E, G, H, I, J, K, O, P, Q, R,S, T, U, V, X, Y, Z, A$, B$, C$, D$
DATA OUTPUT	The string in single quotes (' ') is output to the OUTPUT WINDOW. If a semicolon is added to the end of the string, no line feed is made after output of the string, but a character string or the variable values specified by the next DATA OUTPUT command are output successively.	
DATA OUTPUT @@@@	The value of variable @@@@ is output to the OUTPUT WINDOWwith units added.	@@@@@: E, G, H, I, J, K, O, P, Q, R,S, T, U, V, X, Y, Z, WM, W1, W2, W2-W1, W(CH), LM, L1, L2, L2-L1, L(CH), SPWD, MEANWL, PKWL, PKLVL, MODN, SMSR, SMSR2, WDMCHN, WDMWL(CH), WDMLVL(CH), WDMSNR(CH), NFCHN, NFWL(CH), NFLVLI(CH), NFLVLO(CH), NFASELV(CH), NFGAIN(CH), NFNF(CH), MKPWR, M, N, DOMWL, XCOL, YCOL, ZCOL, CH, A$, B$, C$, D$, FILE$, TIME$
DATA OUTPUT @@@@@;	Outputs the value of variable @@@@@ to the OUTPUT WINDOW with units added. After a string is output, no line feed is sent, but the value of the string or variable of the next DATA OUTPUT command is output.	@@@@@: E, G, H, I, J, K, O, P, Q, R,S, T, U, V, X, Y, Z, WM, W1, W2, W2-W1, W(CH), LM, L1, L2, L2-L1, L(CH), SPWD, MEANWL, PKWL, PKLVL, MODN, SMSR, SMSR2, WDMCHN, WDMWL(CH), WDMLVL(CH), WDMSNR(CH), NFCHN, NFWL(CH), NFLVLI(CH), NFLVLO(CH), NFASELV(CH), NFGAIN(CH), NFNF(CH), MKPWR, M, N, DOMWL, XCOL, YCOL, ZCOL, CH, A$, B$, C$, D$, FILE$, TIME$
DATA OUTPUT DATA AREA	Outputs the contents of the data area to the OUTPUT WINDOW.	
OUTPUT WINDOW CLEAR	Clears the contents of the OUTPUTWINDOW.	
OUTPUT WINDOW ###	Sets whether to display or hide the OUTPUT WINDOW on the screen.	###: ON or OFF

7.5 Controlling an External Instrument with the Program Function

Using the program function, the instrument can remote control the external devices which are connected by various interfaces. In addition, it is possible to remote control the multiple external devices by one program source.

Remote Control of an External Instrument Using the LAN Port

Using the program function, specify the “Computer Name” or “IP address” and “Port Number” of the external device connected to the LAN connector to perform remote control. “Computer Name” or “IP address” must be entered it in the character variable @$ of the program command. If you want to receive query data from the external device, use the send/receive command. Query data is stored in the specified string variable @$.

Send Commands

SEND LAN @$ **** ‘control command (56 chars)’

SEND LAN @$ **** ’control command (56 chars)’ ;@

SEND LAN @$ **** ’control command (20 chars)’ :@:’control command (20 chars)

@$: Computer name or IP address

****: Port Number’

Send/Receive Command

SENDRCV LAN @$ **** ‘query command (56 characters)’

@$: computer name or IP address

****: Port number

Note

- Be sure to set the instrument’s IP address correctly.
- When using DHCP, the instrument’s IP address is automatically set. Set ADDRESS SETTING under TCP/IP SETTING to AUTO (DHCP).
- Please ask your network administrator for details about network connections.
- Using a command such as SEND LAN ’control command (56 characters)’;@, if you insert <wsp> between the command string and the variable @, add “ “ to the end of the command.

7.6 Sample Program

Here, an example is given of performing the operation below.

Conditions

After the measuring conditions have been set, the program performs a single sweep.
Then it searches for a spectrum width and peak wavelength, and outputs the results to the label area and OUTPUT WINDOW. It repeats these operations ten times with a wait of three seconds between repetitions.

```
001     CENTER WL 1555.00nm                 :Set measurement conditions
002     SPAN 10.0nm
003     REFERENCE LEVEL -10.0dBm
004     RESOLUTION 0.1nm
005     AVERAGE TIMES 1
006     SENS NORMAL/HOLD
007     OUTPUT WINDOW CLEAR                 :Clear the OUTPUT WINDOW data.

008     OUTPUT WINDOW ON                    :Display the OUTPUT WINDOW.
009     N=10 :Set loop counter N to 10      :Set loop, counter N to 10
010     SINGLE                              :Perform a single sweep.
011     SPEC WD THRESH 20.0dB               :Perform a spectrum width search
012     DATA OUTPUT 'Wd = ;                 :Output spectrum width to OUTPUT
                                            WINDOW and the label area.
013     LABEL 'Wd = ;'
014     DATA OUTPUT SPWD;
015     LABEL SPWD ;
016     PEAK SEARCH                         :Perform a peak search
017     DATA OUTPUT 'Pk = ;                 :Output the peak wavelength value to
                                            OUTPUT WINDOW and the label area.
018     LABEL 'Pk = ;'
019     DATA OUTPUT PKWL
020     LABEL PKWL
021     WAIT 3S                             :Wait three second.
022     N=N-1 ; IF N <> 0 GOTO 10           :Subtract 1 from loop counter N and if
                                            the result is not 0, make a jump to line
                                            010.
023     END                                 :Exit the Program.
```

Appendix 1 Table of Correspondence between Soft Keys and Remote Commands

The tables below list the remote commands that correspond to the soft keys used when manipulating the various settings of the instrument.

SWEEP

Function	Control Command
Auto	:INITiate:SMODe<wsp>AUTO|3;:INITiate
Repeat	:INITiate:SMODe<wsp>REPeat|2;:INITiate
Single	:INITiate:SMODe<wsp>SINGle|1;:INITiate
Stop	:ABORt
Sweep Marker L1-L2 ON/OFF	:SENSe:WAVelength:SRANge<wsp>OFF|ON|0|1
Sweep Interval *****sec	:SENSe:SWEep:TIME:INTerval<wsp><integer>[SEC]

CENTER

Function	Control Command
Center Wavelength ****.***nm	:SENSe:WAVelength:CENTer<wsp><NRf>[M]
Center Frequency ***.****THz	:SENSe:WAVelength:CENTer<wsp><NRf>[HZ]
Center Wnumber ****.***cm^{-1}	:SENSe:WAVelength:CENTer<wsp><NRf>
Start Wavelength ****.***nm	:SENSe:WAVelength:STARt<wsp><NRf>[M]
Start Frequency ***.****THz	:SENSe:WAVelength:STARt<wsp><NRf>[HZ]
Start Wnumber ****.***cm^{-1}	:SENSe:WAVelength:STARt<wsp><NRf>
Stop Wavelength ****.***nm	:SENSe:WAVelength:STOP<wsp><NRf>[M]
Stop Frequency ***.****THz	:SENSe:WAVelength:STOP<wsp><NRf>[HZ]
Stop Wnumber ****.***cm^{-1}	:SENSe:WAVelength:STOP<wsp><NRf>
Peak WL -> Center	:CALCulate:MARKer:SCENter
Auto Center ON/OFF	:CALCulate:MARKer:MAXimum:SCENter:AUTO<wsp>OFF|ON|0|1
View Scale -> Measure	:DISPlay[:WINDow]:TRACe:X[:SCALe]:SMSCale

SPAN

Function	Control Command
Span Wavelength****.*nm	:SENSe:WAVelength:SPAN<wsp><NRf>[M]
Span Frequency***.**THz	:SENSe:WAVelength:SPAN<wsp><NRf>[HZ]
Span Wnumber****.***cm^{-1}	:SENSe:WAVelength:SPAN<wsp><NRf>
Start Wavelength****.***nm	:SENSe:WAVelength:STARt<wsp><NRf>[M]
Start Frequency***.****THz	:SENSe:WAVelength:STARt<wsp><NRf>[HZ]
Start Wnumber****.***cm^{-1}	:SENSe:WAVelength:STARt<wsp><NRf>
Stop Wavelength****.***nm	:SENSe:WAVelength:STOP<wsp><NRf>[M]
Stop Frequency***.****THz	:SENSe:WAVelength:STOP<wsp><NRf>[HZ]
Stop Wnumber***.****cm^{-1}	:SENSe:WAVelength:STOP<wsp><NRf>
0nm Sweep Time**sec	:SENSe:SWEep:TIME:0NM<wsp><integer>[SEC]
View Scale -> Measure	:DISPlay[:WINDow]:TRACe:X[:SCALe]:SMSCale

LEVEL

Function	Control Command
Reference Level	
Log Scale	:DISPlay:[:WINDow]:TRACe:Y1[:SCAle]:RLEVel<wsp><NRf>[DBM]
Linear Scale	:DISPlay[:WINDow]:TRACe:Y1[:SCALe]:RLEVel<wsp><NRf>[NW|UM|MW]
Log Scale**.*dB/D	:DISPlay[:WINDow]:TRACe:Y1[:SCALe]:PDIVision<wsp><NRf>[DB]
Linear Scale	:DISPlay[:WINDow]:TRACe:Y1[:SCALe]:SPACing<wsp>LINear|1
Linear Base Level**.*mW	:DISPlay[:WINDow]:TRACe:Y1[:SCALe]:BLEVel<wsp><NRf>[MW]
Peak Level -> Ref Level	:CALCulate:MARKer:MAXimum:SRLevel
Auto Ref Level ON/OFF	:CALCulate:MARKer:MAXimum:SRLevel:AUTO
Level Unit dBm / dBm/nm	:DISPlay[:WINDow]:TRACe:Y1[:SCALe]:UNIT<wsp>DBM|DBM/NM
Main Scale Initialize	:DISPlay[:WINDow]:TRACe:Y1[:SCALe]:INITialize
Sub Log**.*dB/D	:DISPlay[:WINDow]:TRACe:Y2[:SCALe]:PDIVision<wsp><NRf>[DB]
Sub Linear*.***/D	:DISPlay[:WINDow]:TRACe:Y2[:SCALe]:PDIVision<wsp><NRf>
Sub Scale**.*dB/km	:DISPlay[:WINDow]:TRACe:Y2[:SCALe]:LENGth
Sub Scale**.*%/D	:DISPlay[:WINDow]:TRACe:Y2[:SCALe]:PDIVision<wsp><NRf>[%]
Offset Level or Scale Minimum **.*dB	:DISPlay[:WINDow]:TRACe:Y2[:SCALe]:OLEVel<wsp><NRf>[DB]
Auto Sub Scale ON/OFF	:DISPlay[:WINDow]:TRACe:Y2[:SCALe]:AUTO<wsp>OFF|ON|0|1
Sub Scale Initialize	:DISPlay[:WINDow]:TRACe:Y2[:SCALe]:INITialize
Y Scale Division 8/10	:DISPlay[:WINDow]:TRACe:Y[:SCALe]:DNUMber<wsp>8|10
Ref Level Position **DIV	:DISPlay[:WINDow]:TRACe:Y1[:SCALe]:RPOSition<wsp><integer>[DIV]
Sub Ref Level Position **DIV	:DISPlay[:WINDow]:TRACe:Y2[:SCALe]:RPOSition<wsp><integer>[DIV]

SETUP

Function	Control Command
Resolution *.***nm	:SENSe:BANDwidth|BWIDth[:RESolution]<wsp><NRf>[M|Hz]
Sensitivity Level **dBm	:SENSe:SENSe:LEVel
Sensitivity	
N/HOLD(x2)	:SENSe:SENSe<wsp>NHLD|0 :SENSe:SWEep:SPEed<wsp>2x|1
N/HOLD	:SENSe:SENSe<wsp>NHLD|0 :SENSe:SWEep:SPEed<wsp>1x|0
N/AUTO(x2)	:SENSe:SENSe<wsp>NAUT|1 :SENSe:SWEep:SPEed<wsp>2x|1
N/AUTO	:SENSe:SENSe<wsp>NAUT|1 :SENSe:SWEep:SPEed<wsp>1x|0
NORMAL(x2)	:SENSe:SENSe<wsp>NORMal|6 :SENSe:SWEep:SPEed<wsp>2x|1
NORMAL	:SENSe:SENSe<wsp>NORMal|6 :SENSe:SWEep:SPEed<wsp>1x|0
MID(x2)	:SENSe:SENSe<wsp>MID|2 :SENSe:SWEep:SPEed<wsp>2x|1
MID	:SENSe:SENSe<wsp>MID|2 :SENSe:SWEep:SPEed<wsp>1x|0
HIGH1(x2)	:SENSe:SENSe<wsp>HIGH1|3 :SENSe:SWEep:SPEed<wsp>2x|1
HIGH1	:SENSe:SENSe<wsp>HIGH1|3 :SENSe:SWEep:SPEed<wsp>1x|0
HIGH2(x2)	:SENSe:SENSe<wsp>HIGH2|4 :SENSe:SWEep:SPEed<wsp>2x|1
HIGH2	:SENSe:SENSe<wsp>HIGH2|4 :SENSe:SWEep:SPEed<wsp>1x|0
HIGH3(x2)	:SENSe:SENSe<wsp>HIGH3|5 :SENSe:SWEep:SPEed<wsp>2x|1
HIGH3	:SENSe:SENSe<wsp>HIGH3|5 :SENSe:SWEep:SPEed<wsp>1x|0
Chop Mode	:SENSe:CHOPper<wsp>OFF|SWITch|0|2
Average Times ***	:SENSe:AVERage:COUNt<wsp><integer>
Sampling Points AUTO	:SENSe:SWEep:POINts:AUTO<wsp>OFF|ON|0|1
Sampling Points *****	:SENSe:SWEep:POINts<wsp><integer>
Sampling Interval *.****nm	:SENSe:SWEep:STEP<wsp><NRf>[M]
Horizontal Scale nm/THz/cm^{-1}	:UNIT:X<wsp>WAVelength|FREQuency|WNUMber|0|1|2
Resolution Correction	:SENSe:SETTing:CORRection<wsp>OFF|ON|0|1|2|MODE1|MODE2
Smoothing ON/OFF	:SENSe:SETTing:SMOothing<wsp>OFF|ON|0|1
Pluse Light Measure	
Peak Hold **msec	:TRIGger[:SEQuence]:STATe<wsp>PHOLd
Ext Trigger Mode	:TRIGger[:SEQuence]:STATe<wsp>ON
Gate Mode	:TRIGger[:SEQuence]:GATE:ITIMe<wsp><NRf>[S]
Gate Setting	
Gate Sampling Interval	:TRIGger[:SEQuence]:GATE:ITIMe<wsp><NRf>[S]
Gate Logic	:TRIGger[:SEQuence]:GATE:LOGic
Measure Delay	:TRIGger[:SEQuence]:GATE:MDELay<wsp><NRf>[S]
Trigger Setting	
Edge Rise/Fall	:TRIGger[:SEQuence]:SLOPe<wsp>RISE|FALL|0|1
Delay ****.*µs	:TRIGger[:SEQuence]:DELay<wsp><NRf>[S]
Trigger Input Mode	:TRIGger[:SEQuence]:INPut
Trigger Output Mode	:TRIGger[:SEQuence]:OUTPut
Fiber Core Size	:SENSe:SETTing:FIBer<wsp>SMALl|LARGe|0|1

ZOOM

Function	Control Command
Zoom Center Wavelength ****.***nm	:DISPlay[:WINDow]:TRACe:X[:SCALe]:CENTer<wsp><NRf>[M]
Zoom Center Frequency ***.****THz	:DISPlay[:WINDow]:TRACe:X[:SCALe]:CENTer<wsp><NRf>[HZ]
Zoom Center Wnumber ****.***cm^{-1}	:DISPlay[:WINDow]:TRACe:X[:SCALe]:CENTer<wsp><NRf>
Zoom Span Wavelength ****.*nm	:DISPlay[:WINDow]:TRACe:X[:SCALe]:SPAN<wsp><NRf>[M]
Zoom Span Frequency ***.**THz	:DISPlay[:WINDow]:TRACe:X[:SCALe]:SPAN<wsp><NRf>[HZ]
Zoom Span Wnumber ****.*cm^{-1}	:DISPlay[:WINDow]:TRACe:X[:SCALe]:SPAN<wsp><NRf>
Zoom Start Wavelength ****.***nm	:DISPlay[:WINDow]:TRACe:X[:SCALe]:STARt<wsp><NRf>[M]
Zoom Start Frequency ***.****THz	:DISPlay[:WINDow]:TRACe:X[:SCALe]:STARt<wsp><NRf>[HZ]
Zoom Start Wnumber ****.***cm^{-1}	:DISPlay[:WINDow]:TRACe:X[:SCALe]:STARt<wsp><NRf>
Zoom Stop Wavelength ****.***nm	:DISPlay[:WINDow]:TRACe:X[:SCALe]:STOP<wsp><NRf>[M]
Zoom Stop Frequency ***.****THz	:DISPlay[:WINDow]:TRACe:X[:SCALe]:STOP<wsp><NRf>[HZ]
Zoom Stop Wnumber ****.***cm^{-1}	:DISPlay[:WINDow]:TRACe:X[:SCALe]:STOP<wsp><NRf>
Peak -> Zoom Ctr	:CALCulate:MARKer:MAXimum:SZCEnter
View Scale -> Measure	:DISPlay[:WINDow]:TRACe:X[:SCALe]:SMSCale
Initialize	:DISPlay[:WINDow]:TRACe:X[:SCALe]:INITialize
Overview Display OFF/L/R	:DISPlay[:WINDow]:OVIew:POSition<wsp>OFF|LEFT|RIGHt|0|1|2

TRACE

Function	Control Command
Active Trace	
A	:TRACe:ACTive<wsp>TRA
B	:TRACe:ACTive<wsp>TRB
C	:TRACe:ACTive<wsp>TRC
D	:TRACe:ACTive<wsp>TRD
E	:TRACe:ACTive<wsp>TRE
F	:TRACe:ACTive<wsp>TRF
G	:TRACe:ACTive<wsp>TRG
View @ DISP/BLANK	:TRACe:STATe:<trace name><wsp>ON|OFF|1|0
Write @	:TRACe:ATTRibute:<trace name><wsp>WRITe|0
Fix @	:TRACe:ATTRibute:<trace name><wsp>FIX|1
Hold @	
Max Hold	:TRACe:ATTRibute:<trace name><wsp>MAX|2
Min Hold	:TRACe:ATTRibute:<trace name><wsp>MIN|3
Roll Average @ ***	:TRACe:ATTRibute:RAVG:<trace name><wsp><integer>
Calculate C@@@@	
Log Math@@@@	
C=A-B(LOG)	:CALCulate:MATH:TRC<wsp>A-B(LOG)
C=B-A(LOG)	:CALCulate:MATH:TRC<wsp>B-A(LOG)
C=A+B(LOG)	:CALCulate:MATH:TRC<wsp>A+B(LOG)
Linear Math@@@@	
C=A+B(LIN)	:CALCulate:MATH:TRC<wsp>A+B(LIN)
C=A-B(LIN)	:CALCulate:MATH:TRC<wsp>A-B(LIN)
C=B-A(LIN)	:CALCulate:MATH:TRC<wsp>B-A(LIN)
C=1-k(A/B) k: *.****	:CALCulate:MATH:TRC:K<wsp><NRf>; :CALCulate:MATH:TRC<wsp>1-K(A/B)
C=1-k(B/A) k: *.****	:CALCulate:MATH:TRC:K<wsp><NRf>; :CALCulate:MATH:TRC<wsp>1-K(B/A)
Calculate F@@@@	
Log Math@@@@	
F=C-D(LOG)	:CALCulate:MATH:TRF<wsp>C-D(LOG)
F=D-C(LOG)	:CALCulate:MATH:TRF<wsp>D-C(LOG)
F=C+D(LOG)	:CALCulate:MATH:TRF<wsp>C+D(LOG)
F=D-E(LOG)	:CALCulate:MATH:TRF<wsp>D-E(LOG)
F=E-D(LOG)	:CALCulate:MATH:TRF<wsp>E-D(LOG)
F=D+E(LOG)	:CALCulate:MATH:TRF<wsp>D+E(LOG)
Linear Math@@@@	
F=C+D(LIN)	:CALCulate:MATH:TRF<wsp>C+D(LIN)
F=C-D(LIN)	:CALCulate:MATH:TRF<wsp>C-D(LIN)
F=D-C(LIN)	:CALCulate:MATH:TRF<wsp>D-C(LIN)
F=D+E(LIN)	:CALCulate:MATH:TRF<wsp>D+E(LIN)
F=D-E(LIN)	:CALCulate:MATH:TRF<wsp>D-E(LIN)
F=E-D(LIN)	:CALCulate:MATH:TRF<wsp>E-D(LIN)
Power/NBW @@@@@@@@@@@@	
F=Pwr/NBW A	:CALCulate:MATH:TRF<wsp>PWRNBWA
F=Pwr/NBW B	:CALCulate:MATH:TRF<wsp>PWRNBWB
F=Pwr/NBW C	:CALCulate:MATH:TRF<wsp>PWRNBWC
F=Pwr/NBW D	:CALCulate:MATH:TRF<wsp>PWRNBWD
F=Pwr/NBW E	:CALCulate:MATH:TRF<wsp>PWRNBWE
Bandwidth	:CALCulate:MATH:TRF:PNBW:BWIDth|BAND
Calculate G@@@@	
Log Math@@@@	
G=C-F(LOG)	:CALCulate:MATH:TRG<wsp>C-F(LOG)

Function	Control Command
G=F-C(LOG)	:CALCulate:MATH:TRG<wsp>F-C(LOG)
G=C+F(LOG)	:CALCulate:MATH:TRG<wsp>C+F(LOG)
G=E-F(LOG)	:CALCulate:MATH:TRG<wsp>E-F(LOG)
G=F-E(LOG)	:CALCulate:MATH:TRG<wsp>F-E(LOG)
G=E+F(LOG)	:CALCulate:MATH:TRG<wsp>E+F(LOG)
Linear Math@@@@	
G=C+F(LIN)	:CALCulate:MATH:TRG<wsp>C+F(LIN)
G=C-F(LIN)	:CALCulate:MATH:TRG<wsp>C-F(LIN)
G=F-C(LIN)	:CALCulate:MATH:TRG<wsp>F-C(LIN)
G=E+F(LIN)	:CALCulate:MATH:TRG<wsp>E+F(LIN)
G=E-F(LIN)	:CALCulate:MATH:TRG<wsp>E-F(LIN)
G=F-E(LIN)	:CALCulate:MATH:TRG<wsp>F-E(LIN)
Normalize@@@@	
G=NORM A	:CALCulate:MATH:TRG<wsp>NORMA
G=NORM B	:CALCulate:MATH:TRG<wsp>NORMB
G=NORM C	:CALCulate:MATH:TRG<wsp>NORMC
Curve Fit@@@@	
G=CRV FIT A	:CALCulate:MATH:TRG<wsp>CVFTA
G=CRV FIT B	:CALCulate:MATH:TRG<wsp>CVFTB
G=CRV FIT C	:CALCulate:MATH:TRG<wsp>CVFTC
G=MKR FIT	:CALCulate:MATH:TRG<wsp>MKRFT
Threshold **dB	:CALCulate:MATH:TRG:CVFT:THResh<wsp><NRf>[DB]
Operation Area	:CALCulate:MATH:TRG:CVFT:OPARea<wsp>ALL|INL1-L2|OUTL1-L2|0|1|2
Fitting Algorithm	:CALCulate:MATH:TRG:CVFT:FALGo<wsp>GAUSS|LORENz|3RD|4TH|5TH|0|1|2|3|4
Peak Curve Fit @@@@	
G=PKCVFIT A	:CALCulate:MATH:TRG<wsp>PKCVFTA
G=PKCVFIT B	:CALCulate:MATH:TRG<wsp>PKCVFTB
G=PKCVFIT C	:CALCulate:MATH:TRG<wsp>PKCVFTC
Threshold **dB	:CALCulate:MATH:TRG:PCVFt:THResh<wsp><NRf>[DB]
Operation Area	:CALCulate:MATH:TRG:CVFT:OPARea<wsp>ALL|INL1-L2|OUTL1-L2|0|1|2
Fitting Algorithm	:CALCulate:MATH:TRG:CVFT:FALGo<wsp>GAUSS|LORENz|3RD|4TH|5TH|0|1|2|3|4
Trace List	
Trace Copy	:TRACe:COPY<wsp><source trace name>,<destination trace name>
Trace Clear	:TRACe:DELete<wsp><trace name>
Label	:DISPlay[:WINDow]:TEXT:DATA<wsp><string>
Noise Mask ***dB	:DISPlay[:WINDow]:TRACe:Y:NMASk<wsp><NRf>[DB]
Mask Line VERT / HRZN	:DISPlay[:WINDow]:TRACe:Y:NMASk:TYPE<wsp>VERTical|HORIzontal|0|1

MARKER

Function	Control Command
Marker Active ON/OFF	:CALCulate:MARKer[:STATe]<wsp><marker>,|ON|1
Set Marker	:CALCulate:MARKer:X<wsp><marker>,<NRf>[M|HZ]
Clear Marker	:CALCulate:MARKer[:STATe]<wsp><marker>,OFF|0
Marker Setting	:CALCulate:MARKer[:STATe]<wsp><marker>,OFF|ON||DENSity|INTegral|0|1|2|3[,<integra range>]
Normal Marker	:CALCulate:MARKer[:STATe]<wsp><marker>ON|1
Power Density	:CALCulate:MARKer[:STATe]<wsp><marker>DENSity|2
Integral Power	:CALCulate:MARKer[:STATe]<wsp><marker>INTegral|3
Bandwidth	:CALCulate:MARKer{:PDENsity|:NOISe}{:BWIDth|:BANDwidth}<wsp><NRf>[m]
Marker -> Center	:CALCulate:MARKer:SCENter
Marker -> Zoom Ctr	:CALCulate:MARKer:SZCenter
Marker -> Ref Level	:CALCulate:MARKer:SRLevel
All Marker Clear	:CALCulate:MARKer:AOFF
Line Marker 1 ON/OFF	:CALCulate:LMARker:X<wsp>1,<NRf>[M]
Line Marker 2 ON/OFF	:CALCulate:LMARker:X<wsp>2,<NRf>[M]
Line Marker 3 ON/OFF	:CALCulate:LMARker:Y<wsp>3,<NRf>[DBM]
Line Marker 4 ON/OFF	:CALCulate:LMARker:Y<wsp>4,<NRf>[DBM]
Marker L1-L2 -> Span	:CALCulate:LMARker:SSPan
Marker L1-L2 -> Zoom Span	:CALCulate:LMARker:SZSPan
Line Marker All Clear	:CALCulate:LMARker:AOFF
Marker Display	
Offset	:CALCulate:MARKer:FUNCtion:FORMat<wsp>OFFSet|0
Spacing	:CALCulate:MARKer:FUNCtion:FORMat<wsp>SPACing|1
Marker Auto Update ON/OFF	:CALCulate:MARKer:FUNCtion:UPDate<wsp>OFF|ON|0|1
Marker Unit nm/THz	:CALCulate:MARKer:UNIT<wsp>WAVelength|FREQuency|0|1
Search/Ana Marker L1-L2 ON/OFF	:CALCulate:LMARker:SRANge<wsp>OFF|ON|0|1
Search/Ana Zoom Area ON/OFF	:DISPlay[:WINDow]:TRACe:X[:SCALe]:SRANge<wsp>OFF|ON|0|1
Sweep Marker L1-L2 ON/OFF	:SENSe:WAVelength:SRANge<wsp>OFF|ON|0|1

PEAK SEARCH

Function	Control Command
Peak Search	:CALCulate:MARKer:MAXimum
Bottom Search	:CALCulate:MARKer:MINimum
Next Level Search	:CALCulate:MARKer:MAXimum:NEXT or :CALCulate:MARKer:MINimum:NEXT
Next Search Right	:CALCulate:MARKer:MAXimum:RIGHt or :CALCulate:MARKer:MINimum:RIGHt
Next Search Left	:CALCulate:MARKer:MAXimum:LEFT or :CALCulate:MARKer:MINimum:LEFT
Set Marker	:CALCulate:MARKer[:STATe]<wsp><marker>,ON|1
Clear Marker	:CALCulate:MARKer[:STATe]<wsp><marker>,OFF|0
All Marker Clear	:CALCulate:MARKer:AOFF
Auto Search ON/OFF	:CALCulate:MARKer:AUTO<wsp>OFF|ON|0|1
Mode Diff **.**dB	:CALCulate:PARameter:COMMon:MDIFf<wsp><NRf>[DB]
Search/Ana Marker L1-L2 ON/OFF	:CALCulate:LMARker:SRANge<wsp>OFF|ON|0|1
Search/Ana Zoom Area ON/OFF	:DISPlay[:WINDow]:TRACe:X[:SCALe]:SRANge<wsp> OFF|ON|0|1
Search Mode SINGL/MULTI	:CALCulate:MARKer:MSEarch<wsp>OFF|ON|0|1
Multi Search Setting	
Threshold **.**dB	:CALCulate:MARKer:MSEarch:THResh<wsp><NRf>[DB]
Sort By WL/LVL	:CALCulate:MARKer:MSEarch:SORT<wsp> WAVelength|LEVel|0|1

ANALYSIS

Function	Control Command
Spec Width@@@@	
THRESH	:CALCulate:CATegory<wsp>SWTHresh|0
ENVELOPE	:CALCulate:CATegory<wsp>SWENvelope|1
RMS	:CALCulate:CATegory<wsp>SWRMs|2
PEAK RMS	:CALCulate:CATegory<wsp>SWPKrms|3
NOTCH	:CALCulate:CATegory<wsp>NOTCh|4
Analysis1@@@@	
DFB-LD	:CALCulate:CATegory<wsp>DFBLd|5
FP-LD	:CALCulate:CATegory<wsp>FPLD|6
LED	:CALCulate:CATegory<wsp>LED|7
SMSR	:CALCulate:CATegory<wsp>SMSR|8
POWER	:CALCulate:CATegory<wsp>POWer|9
TLS	:CALCulate:CATegory<wsp>ITLa|18
Analysis2@@@@@	
WDM	:CALCulate:CATegory<wsp>WDM|11
EDFA-NF	:CALCulate:CATegory<wsp>NF|12
FILTER-PK	:CALCulate:CATegory<wsp>FILPk|13
FILTER-BTM	:CALCulate:CATegory<wsp>FILBtm|14
WDM FIL-PK	:CALCulate:CATegory<wsp>WFPeak|15
WDM FIL-BTM	:CALCulate:CATegory<wsp>WFBtm|16
COLOR	:CALCulate:CATegory<wsp>COLor|17
Analysis Execute (@@@@)	:CALCulate[:IMMediate]
Spec Width Thresh **.*dB	:CALCulate:PARameter[:CATegory]:SWTHresh:TH<wsp><NRf>[DB]
Switch Display	
Trace & Table	:CALCulate:DISPlay<wsp>0
Table	:CALCulate:DISPlay<wsp>1
Trace	:CALCulate:DISPlay<wsp>2
Graph & Table	:CALCulate:DISPlay<wsp>3
Graph	:CALCulate:DISPlay<wsp>4
Auto Analysis ON/OFF	:CALCulate[:IMMediate]:AUTO<wsp>OFF|ON|0|1
Search/Ana Marker L1-L2 ON/OFF	:CALCulate:LMARker:SRANge<wsp>
Search/Ana Zoom Area ON/OFF	:DISPlay[:WINDow]:TRACe:X[:SCALe]:SRANge<wsp>OFF|ON|0|1
Grid Setting	
Start Wavelength ****.****nm	:SYSTem:GRID:CUSTom:STARt<wsp><NRf>[M|HZ]
Stop Wavelength ****.****nm	:SYSTem:GRID:CUSTom:STOP<wsp><NRf>[M|HZ]
Spacing ***.**GHz	:SYSTem:GRID:CUSTom:SPACing<wsp><NRf>[GHZ]
Reference Frequency ****.****THz	:SYSTem:GRID:REFerence<wsp><NRf>[HZ]

FILE

Function	Control Command
Save	
File Name	:MMEMory:CDIRectory<wsp><directory name>
Item Select	
Trace	:MMEMory:STORe:TRACe<wsp><trace name>,BIN|CSV,<"file name">[,INTernal|EXTernal]
All Trace	:MMEMory:STORe:ATRace<wsp><"file name"> [,INTernal|EXTernal]
Graphics	:MMEMory:STORe:GRAPhics<wsp>B&W|COLor|PCOLor,BMP|PNG|JPG,<"file name">[,INTernal|EXTernal]
Setting	:MMEMory:STORe:SETTing<wsp><"file name"> [,INTernal|EXTernal]
Data	:MMEMory:STORe:DATA<wsp><"file name"> [,INTernal|EXTernal]
Output Item Setting	
Date & Time ON/OFF	:MMEMory:STORe:DATA:ITEM<wsp>DATE,OFF|ON|0|1
Label ON/OFF	:MMEMory:STORe:DATA:ITEM<wsp>LABel,OFF|ON|0|1
Data Area ON/OFF	:MMEMory:STORe:DATA:ITEM<wsp>DATA,OFF|ON|0|1
Condition ON/OFF	:MMEMory:STORe:DATA:ITEM<wsp>CONDition,OFF|ON|0|1
Trace Data ON/OFF	:MMEMory:STORe:DATA:ITEM<wsp>TRACe,OFF|ON|0|1
Output Window ON/OFF	:MMEMory:STORe:DATA:ITEM<wsp>OWINdow,OFF|ON|0|1
Write Mode ADD/OVER	:MMEMory:STORe:DATA:MODE<wsp>ADD|OVER|0|1
Auto File Name	:MMEMory:ANAMe<wsp>NUMBer|DATE
Remove USB Storage	:MMEMory:REMove
File Operation	
Memory INT/EXT	:MMEMory:CDRive<wsp>INTernal|EXTernal
Delete	:MMEMory:DELete<wsp><"file name"> [,INTernal|EXTernal]
Copy	:MMEMory:COPY<wsp><"source file name">,INTernal|EXTernal,<"destination file name> [,INTernal|EXTernal]
Rename	:MMEMory:RENName<wsp><"new file name">, <"old file name">[,INTernal|EXTernal]
Make Directory	:MMEMory:MDIRectory<wsp><"directory name">[,INTernal|EXTernal]

SYSTEM

Function	Control Command
Optical Alignment	:CALibration:ALIGn:INTernal[:IMMediate]
Wavelength Calibration	
Built-in Source	:CALibration:WAVelength:INTernal[:IMMediate]
External Laser ****.***nm	:CALibration:WAVelength:EXTernal:SOURce<wsp>LASer|0; :CALibration:WAVelength:EXTernal:WAVelength<wsp><NRf>[M]
External Gas Cell ****.***nm	:CALibration:WAVelength:EXTernal:SOURce<wsp>GASCell|1; :CALibration:WAVelength:EXTernal:WAVelength<wsp><NRf>[M]
Wavelength Shift **.***nm	:SENSe:CORRection:WAVelength:SHIFt<wsp><NRf>[M]
Level Shift ***.***dB	:SENSe:CORRection:LEVel:SHIFt<wsp><NRf>[DB]
Wavelength in Air/Vacuum	:SENSe:CORRection:RVELocity:MEDium<wsp>AIR|VACuum|0|1
Command Format	:SYSTem:COMMunicate:CFORmat<wsp>AQ6317|AQ6374E|0|1
Monitor Port ON/OFF	:SYSTem:COMMunicate:RMONitor<wsp>OFF|ON|0|1
Auto Offset Setting	
Auto Offset ON/OFF	:CALibration:ZERO[:AUTO]<wsp>OFF|ON|0|1
Interval ***min	:CALibration:ZERO[:AUTO]:INTerval<wsp><integer>
Uncal Warning ON/OFF	:SYSTem:DISPlay:UNCal<wsp>OFF|ON|0|1
Buzzer	
Click ON/OFF	:SYSTem:BUZZer:CLICk<wsp>OFF|ON|0|1
Warning ON/OFF	:SYSTem:BUZZer:WARNing<wsp>OFF|ON|0|1
Level Display Digit	
1Digit	:UNIT:POWer:DIGit<wsp>1
2Digit	:UNIT:POWer:DIGit<wsp>2
3Digit	:UNIT:POWer:DIGit<wsp>3
Color Mode	
COLOR	:DISPlay:COLor<wsp>1
B&W	:DISPlay:COLor<wsp>0
Set Clock	:SYSTem:DATE<wsp><year>,<month>,<day>
	:SYSTem:TIME<wsp><hour>,<minutes>,<seconds>
User Key Define	–
Operation Lock	:SYSTem:OLOCk
System Information	:SYSTem:INFormation?<wsp>0|1
Parameter Initialize	
Parameter Clear	:SYSTem:PRESet
Display Off	:DISPlay[:WINDow]<wsp>OFF|ON|0|1

PRESET

Function	Control Command
PRESET	:SYSTem:PRESet

APP

Function	Control Command
Data Logging	
Start/Stop	:APPLication:DLOGging:STATe<wsp>STOP|STARt|0|1
Setup	
Logging Parameter	
LOGGING SETTING	
LOGGING ITEM	:APPLication:DLOGging:LPARameter:ITEM<wsp>0|1|2|3
LOGGING MODE	:APPLication:DLOGging:LPARameter:LMODe<wsp>1|2
MINIMUM INTERVAL	:APPLication:DLOGging:LPARameter:INTerval<wsp><integer>[SEC]
TEST DURATION	:APPLication:DLOGging:LPARameter:TDURation<wsp><integer>[sec]
PEAK DETECTION SETTING	
PEAK THRESH TYPE	:APPLication:DLOGging:LPARameter:PDETect:TTYPe<wsp>ABSolute|RELative
THRESH(ABS)	:APPLication:DLOGging:LPARameter:PDETect:ATHResh<NRf>[DBM]
THRESH(REL)	:APPLication:DLOGging:LPARameter:PDETect:RTHResh<NRf>[DB]
CH MATCHING λ THRESH	:APPLication:DLOGging:LPARameter:MTHResh<wsp><NRf>[M]
TRACE DATA SAVE SETTING	
TRACE LOGGING	:APPLication:DLOGging:LPARameter:TLOGging<wsp>OFF|ON|0|1
DESTINATION MEMORY	:APPLication:DLOGging:LPARameter:MEMory<wsp>INTernal|EXTernal
Logging Data Save	:MMEMory:STORe:DLOGging<wsp><"file name">[,INTernal|EXTernal]
Logging Data Load	:MMEMory:LOAD:DLOGging<wsp><"file name">[,INTernal|EXTernal]
PROGRAM Execute	:APPLication:PROGram:EXECute

Appendix 2 ANALYSIS Setting Parameters

In setting ANALYSIS key setting parameters, the analysis parameters differ with the analysis type. Thus, the PARAMETER SETTING key commands are set independently of the regular key commands. An analysis parameter setting command is shown below.

Spec Width

ANALYSIS Parameters	Control Command
THRESH	
THRESH LEVEL **.**dB	:CALCulate:PARameter[:CATegory]:SWTHresh:TH<wsp><NRf>[DB]
K **.**	:CALCulate:PARameter[:CATegory]:SWTHresh:K<wsp><NRf>
MODE FIT ON/OFF	:CALCulate:PARameter[:CATegory]:SWTHresh:MFIT<wsp> OFF|ON|0|1
ENVELOPE	
THRESH LEVEL1**.**dB	:CALCulate:PARameter[:CATegory]:SWENvelope:TH1<wsp> <NRf>[DB]
THRESH LEVEL2 **.**dB	:CALCulate:PARameter[:CATegory]:SWENvelope:TH2<wsp> <NRf>[DB]
K **.**	:CALCulate:PARameter[:CATegory]:SWENvelope:K<wsp><NRf>
RMS	
THRESH LEVEL **.**dB	:CALCulate:PARameter[:CATegory]:SWRMs:TH<wsp><NRf>[DB]
K **.**	:CALCulate:PARameter[:CATegory]:SWRMs:K<wsp><NRf>
PEAK RMS	
THRESH LEVEL **.**dB	:CALCulate:PARameter[:CATegory]:SWPKrms:TH<wsp><NRf>[DB]
K **.**	:CALCulate:PARameter[:CATegory]:SWPKrms:K<wsp><NRf>
NOTCH	
THRESH LEVEL **.**dB	:CALCulate:PARameter[:CATegory]:NOTCh:TH<wsp><NRf>[DB]
K **.**	:CALCulate:PARameter[:CATegory]:NOTCh:K<wsp><NRf>
Type	
PEAK	:CALCulate:PARameter[:CATegory]:NOTCh:TYPE<wsp>PEAK|0
BOTTOM	:CALCulate:PARameter[:CATegory]:NOTCh:TYPE<wsp>BOTTom|1

Analysis 1

ANALYSIS Parameters	Control Command
DFB-LD	
-XdB CENTER / WIDTH	
ALGO	:CALCulate:PARameter[:CATegory]:DFBLd<wsp>SWIDth,ALGO,<data>
THRESH **.**dB	:CALCulate:PARameter[:CATegory]:DFBLd<wsp> SWIDth,TH,<NRf>[DB]
THRESH2 **.**dB	:CALCulate:PARameter[:CATegory]:DFBLd<wsp>SWIDth,TH2,<NRf>[DB]
K	:CALCulate:PARameter[:CATegory]:DFBLd<wsp>SWIDth,K,<NRf>
MODE FIT ON/OFF	:CALCulate:PARameter[:CATegory]:DFBLd<wsp>SWIDth,MFIT,OFF|ON|0|1
MODE DIFF *.**dB	:CALCulate:PARameter[:CATegory]:DFBLd<wsp>SWIDth,MDIFf,<NRf>[DB]
DFB-LD	
SMSR	
SMSR MODE	:CALCulate:PARameter[:CATegory]:DFBLd<wsp>SMSR,SMODe,SMSR1|SMSR2|SMSR3|SMSR4
SMSR MASK ±*.**nm	:CALCulate:PARameter[:CATegory]:DFBLd<wsp>SMSR,SMASk,<NRf>[M]
MODE DIFF *.**dB	:CALCulate:PARameter[:CATegory]:DFBLd<wsp>SMSR,MDIFf,<NRf>[DB]
SIDE MODE POWER	:CALCulate:PARameter[:CATegory]:SMSR:SMPower<wsp>0|1|TDATa|NORMalized
BAND WIDTH	:CALCulate:PARameter[:CATegory]:SMSR:BANDwidth|:BWIDth<wsp><nrf>[nm]
RMS	
ALGO	:CALCulate:PARameter[:CATegory]:DFBLd<wsp>RMS,ALGO,<data>
THRESH **.**dB	:CALCulate:PARameter[:CATegory]:DFBLd<wsp>RMS,TH,<NRf>[DB]
K	:CALCulate:PARameter[:CATegory]:DFBLd<wsp>RMS,K,<NRf>
MODE DIFF *.**dB	:CALCulate:PARameter[:CATegory]:DFBLd<wsp>RMS,MDIFf,<NRf>[DB]
POWER	
SPAN **.**nm	:CALCulate:PARameter[:CATegory]:DFBLd<wsp>POWer,SPAN,<NRf>[M]
OSNR	
MODE DIFF *.**dB	:CALCulate:PARameter[:CATegory]:DFBLd<wsp>OSNR,MDIFf,<NRf>[DB]
NOISE ALGO	:CALCulate:PARameter[:CATegory]:DFBLd<wsp>OSNR,NALGo,<data>
NOISE AREA **.**nm	:CALCulate:PARameter[:CATegory]:DFBLd<wsp>OSNR,NAREa,<NRf>[M]
MASK AREA **.**nm	:CALCulate:PARameter[:CATegory]:DFBLd<wsp>OSNR,MAREa,<NRf>[M]
FITTING ALGO	:CALCulate:PARameter[:CATegory]:DFBLd<wsp>OSNR,FALGo,<data>
NOISE BW *.**nm	:CALCulate:PARameter[:CATegory]:DFBLd<wsp>OSNR,NBW,<NRf>[M]
SIGNAL POWER	:CALCulate:PARameter[:CATegory]:DFBLd<wsp>OSNR,SPOWer,<data>
INTEGRAL RANGE	:CALCulate:PARameter[:CATegory]:DFBLd<wsp>OSNR,IRANge,<NRf>

ANALYSIS Parameters	Control Command
FP-LD	
SPECTRUM WIDTH	
ALGO	:CALCulate:PARameter[:CATegory]:FPLD<wsp>SWIDth,ALGO,<data>
THRESH **.**dB	:CALCulate:PARameter[:CATegory]:FPLD<wsp>SWIDth,TH,<NRf>[DB]
THRESH2 **.**dB	:CALCulate:PARameter[:CATegory]:FPLD<wsp>SWIDth,TH2,<NRf>[DB]
K	:CALCulate:PARameter[:CATegory]:FPLD<wsp>SWIDth,K,<NRf>
MODE FIT ON/OFF	:CALCulate:PARameter[:CATegory]:FPLD<wsp>SWIDth,MFIT,OFF|ON|0|1
MODE DIFF *.**dB	:CALCulate:PARameter[:CATegory]:FPLD<wsp>SWIDth,MDIFf,<NRf>[DB]
MEAN WAVELENGTH	
ALGO	:CALCulate:PARameter[:CATegory]:FPLD<wsp>MWAVelength,ALGO,<data>
THRESH **.**dB	:CALCulate:PARameter[:CATegory]:FPLD<wsp>MWAVelength,TH,<NRf>[DB]
THRESH2 **.**dB	:CALCulate:PARameter[:CATegory]:FPLD<wsp>MWAVelength,TH2,<NRf>[DB]
K	:CALCulate:PARameter[:CATegory]:FPLD<wsp>MWAVelength,K,<NRf>
MODE FIT ON/OFF	:CALCulate:PARameter[:CATegory]:FPLD<wsp>MWAVelength,MFIT,OFF|ON|0|1
MODE DIFF *.**dB	:CALCulate:PARameter[:CATegory]:FPLD<wsp>MWAVelength,MDIFf,<NRf>[DB]
TOTAL POWER	
OFFSET LEVEL *.**dB	:CALCulate:PARameter[:CATegory]:FPLD<wsp>TPOWer,OFFSet,<NRf>[DB]
MODE NO.	
ALGO	:CALCulate:PARameter[:CATegory]:FPLD<wsp>MNUMber,ALGO,<data>
THRESH **.**dB	:CALCulate:PARameter[:CATegory]:FPLD<wsp>MNUMber,TH,<NRf>[DB]
THRESH2 **.**dB	:CALCulate:PARameter[:CATegory]:FPLD<wsp>MNUMber,TH2,<NRf>[DB]
K	:CALCulate:PARameter[:CATegory]:FPLD<wsp>MNUMber,K,<NRf>
MODE FIT ON/OFF	:CALCulate:PARameter[:CATegory]:FPLD<wsp>MNUMber,MFIT,OFF|ON|0|1
MODE DIFF *.**dB	:CALCulate:PARameter[:CATegory]:FPLD<wsp>MNUMber,MDIFf,<NRf>[DB]
LED	
SPECTRUM WIDTH	
ALGO	:CALCulate:PARameter[:CATegory]:LED<wsp>SWIDth,ALGO,<data>
THRESH **.**dB	:CALCulate:PARameter[:CATegory]:LED<wsp>SWIDth,TH,<NRf>[DB]
THRESH2 **.**dB	:CALCulate:PARameter[:CATegory]:LED<wsp>SWIDth,TH2,<NRf>[DB]
K	:CALCulate:PARameter[:CATegory]:LED<wsp>SWIDth,K,<NRf>
MODE FIT ON/OFF	:CALCulate:PARameter[:CATegory]:LED<wsp>SWIDth,MFIT,OFF|ON|0|1
MODE DIFF *.**dB	:CALCulate:PARameter[:CATegory]:LED<wsp>SWIDth,MDIFf,<NRf>[DB]

ANALYSIS Parameters	Control Command
MEAN WAVELENGTH	
ALGO	:CALCulate:PARameter[:CATegory]:LED<wsp>MWAVelength,ALGO,<data>
THRESH **.**dB	:CALCulate:PARameter[:CATegory]:LED<wsp>MWAVelength,TH,<NRf>[DB]
THRESH2 **.**dB	:CALCulate:PARameter[:CATegory]:LED<wsp>MWAVelength,TH2,<NRf>[DB]
K	:CALCulate:PARameter[:CATegory]:LED<wsp>MWAVelength,K,<NRf>
MODE FIT ON/OFF	:CALCulate:PARameter[:CATegory]:LED<wsp>MWAVelength,MFIT,OFF|ON|0|1
MODE DIFF *.**dB	:CALCulate:PARameter[:CATegory]:LED<wsp>MWAVelength,MDIFf,<NRf>[DB]
TOTAL POWER	
OFFSET LEVEL *.**dB	:CALCulate:PARameter[:CATegory]:LED<wsp>TPOWer,OFFSet,<NRf>[DB]
SMSR	
SMSR MODE	:CALCulate:PARameter[:CATegory]:SMSR:MODE<wsp>SMSR1|SMSR2|SMSR3|SMSR4
SMSR MASK ±*.**nm	:CALCulate:PARameter[:CATegory]:SMSR:MASK<wsp><NRf>[M]
SIDE MODE POWER	:CALCulate:PARameter[:CATegory]:SMSR:SMPower<wsp>0|1|TDATa|NORMalized
BANDWITTH *.**nm	:CALCulate:PARameter[:CATegory]:SMSR:BANDwidth|:BWIDth<wsp><nrf>[nm]
MODE DIFF *.**dB	:CALCulate:PARameter[:CATegory]:SMSR:MDIFf<wsp><NRf>[DB]
POWER	
POWER OFFSET *.**dB	:CALCulate:PARameter[:CATegory]:POWer:OFFSet<wsp><NRf>[DB]
TLS	
-XdB CENTER / WIDTH	
ALGO	:CALCulate:PARameter[:CATegory]:ITLa<wsp>SWIDth,ALGO,ENVelope|THResh|RMS|PKRMs
THRESH **.**dB	:CALCulate:PARameter[:CATegory]:ITLa<wsp>SWIDth,TH,<NRf>[DB]
THRESH2 **.**dB	:CALCulate:PARameter[:CATegory]:ITLa<wsp>SWIDth,TH2,<NRf>[DB]
K	:CALCulate:PARameter[:CATegory]:ITLa<wsp>SWIDth,K,<NRf>
MODE FIT ON/OFF	:CALCulate:PARameter[:CATegory]:ITLa<wsp>SWIDth,MFIT,OFF|ON|0|1
MODE DIFF *.**dB	:CALCulate:PARameter[:CATegory]:ITLa<wsp>SWIDth,MDIFf,<NRf>[DB]
SMSR	
SMSR MODE	:CALCulate:PARameter[:CATegory]:ITLa<wsp>SMSR,SMODe,SMSR1|SMSR2|SMSR3|SMSR4
SMSR MASK ± *.**nm	:CALCulate:PARameter[:CATegory]:ITLa<wsp>SMSR,SMASk,<NRf>[M]
MODE DIFF *.**dB	:CALCulate:PARameter[:CATegory]:ITLa<wsp>SMSR,MDIFf,<NRf>[DB]
SIDE MODE POWER	:CALCulate:PARameter[:CATegory]:SMSR:SMPower<wsp>0|1|TDATa|NORMalized
BAND WIDTH	:CALCulate:PARameter[:CATegory]:SMSR:BANDwidth|:BWIDth<wsp><nrf>[nm]
POWER	
SPAN	:CALCulate:PARameter[:CATegory]:ITLa<wsp>POWer,SPAN,<NRf>[M]

ANALYSIS Parameters	Control Command
SSER / STSSER	
ALGO	:CALCulate:PARameter[:CATegory]:ITLa<wsp>SSER,ALGO,IEC|CURVefit
MODE DIFF	:CALCulate:PARameter[:CATegory]:ITLa<wsp>SSER,MDIFf,<NRf>[DB]
ANALYSIS AREA	:CALCulate:PARameter[:CATegory]:ITLa<wsp>SSER,AAREa,<NRf>[M]
FITTING AREA	:CALCulate:PARameter[:CATegory]:ITLa<wsp>SSER,FAREa,<NRf>[M]
MASK AREA	:CALCulate:PARameter[:CATegory]:ITLa<wsp>SSER,MAREa,<NRf>[M]
FITTING ALGO	:CALCulate:PARameter[:CATegory]:ITLa<wsp>SSER,FALGo,GAUSs|LORenz|3RD|4TH|5TH|1|2|3|4|5
NOISE BW *.**nm	:CALCulate:PARameter[:CATegory]:ITLa<wsp>SSER,NBW,<NRf>[M]
SIGNAL POWER	:CALCulate:PARameter[:CATegory]:ITLa<wsp>SSER,SPOWer,PEAK|INTegral|0|1
INTEGRAL RANGE	:CALCulate:PARameter[:CATegory]:ITLa<wsp>SSER,IRANge,<NRf>

Analysis 2

ANALYSIS Parameters	Control Command
WDM	
CHANNEL DETECTION SETTING	
THRESH LEVEL	:CALCulate:PARameter[:CATegory]:WDM:TH<wsp><NRf>[DB]
MODE DIFF **.**dB	:CALCulate:PARameter[:CATegory]:WDM:MDIFf<wsp><NRf>[DB]
DISPLAY MASK OFF/ON *.**dB	:CALCulate:PARameter[:CATegory]:WDM:DMASk<wsp><NRf>[DB]
INTERPOLATATION SETTING	
NOISE ALGO	
AUTO-FIX	:CALCulate:PARameter[:CATegory]:WDM:NALGo<wsp>AFIX|0
MANUAL-FIX	:CALCulate:PARameter[:CATegory]:WDM:NALGo<wsp>MFIX|1
AUTO-CTR	:CALCulate:PARameter[:CATegory]:WDM:NALGo<wsp>ACENter|2
MANUAL-CTR	:CALCulate:PARameter[:CATegory]:WDM:NALGo<wsp>MCENter|3
PIT	:CALCulate:PARameter[:CATegory]:WDM:NALGo<wsp>PIT|4
NOISE AREA	:CALCulate:PARameter[:CATegory]:WDM:NAREa<wsp><NRf>[M]
MASK AREA	:CALCulate:PARameter[:CATegory]:WDM:MAREa<wsp><NRf>[M]
FITTING ALGO	
LINEAR	:CALCulate:PARameter[:CATegory]:WDM:FALGo<wsp>LINear|0
GAUSS	:CALCulate:PARameter[:CATegory]:WDM:FALGo<wsp>GAUSs|1
LORENZ	:CALCulate:PARameter[:CATegory]:WDM:FALGo<wsp>LORenz|2
3RD POLY	:CALCulate:PARameter[:CATegory]:WDM:FALGo<wsp>3RD|3
4TH POLY	:CALCulate:PARameter[:CATegory]:WDM:FALGo<wsp>4TH|4
5TH POLY	:CALCulate:PARameter[:CATegory]:WDM:FALGo<wsp>5TH|5
NOISE BW *.**nm	:CALCulate:PARameter[:CATegory]:WDM:NBW<wsp><NRf>[M]
DUAL TRACE ON/OFF	:CALCulate:PARameter[:CATegory]:WDM:DUAL<wsp>OFF|ON|0|1
DISPLAY SETTING	
DISPLAY TYPE	
ABSOLUTE	:CALCulate:PARameter[:CATegory]:WDM:DTYPe<wsp>ABSolute |0
RELATIVE	:CALCulate:PARameter[:CATegory]:WDM:DTYPe<wsp>RELatibe|1
DRIFT(MEAS)	:CALCulate:PARameter[:CATegory]:WDM:DTYPe<wsp>MDRift|2
DRIFT(GRID)	:CALCulate:PARameter[:CATegory]:WDM:DTYPe<wsp>GDRift|3
CH RELATION	
OFFSET	:CALCulate:PARameter[:CATegory]:WDM:RELation<wsp>OFFSet |0
SPACING	:CALCulate:PARameter[:CATegory]:WDM:RELation<wsp>SPACing|1
REF CH	:CALCulate:PARameter[:CATegory]:WDM:RCH<wsp><integer>
MAX/MIN RESET	:CALCulate:PARameter[:CATegory]:WDM:MMReset
OUTPUT SLOPE ON/OFF	:CALCulate:PARameter[:CATegory]:WDM:OSLope<wsp>OFF|ON|0|1
POINT DISPLAY ON/OFF	:CALCulate:PARameter[:CATegory]:WDM:PDISplay<wsp>OFF|ON|0|1
OTHER SETTING	
SIGNAL POWER	:CALCulate:PARameter[:CATegory]:WDM:SPOWer<wsp>PEAK|INTegral|0|1
INTEGRAL RANGE	:CALCulate:PARameter[:CATegory]:WDM:IRANge<wsp><NRf>
EDFA-NF	
CHANNNEL DETECTION SETTING	
THRESH LEVEL **.**dB	:CALCulate:PARameter[:CATegory]:NF:TH<wsp><NRf>[DB]
MODE DIFF **.**dB	:CALCulate:PARameter[:CATegory]:NF:MDIFf<wsp><NRf>[DB]
INTERPOLATION SETTING	
OFFSET(IN) **.**dB	:CALCulate:PARameter[:CATegory]:NF:IOFFset<wsp><NRf>[DB]
OFFSET(OUT) **.**dB	:CALCulate:PARameter[:CATegory]:NF:OOFFset<wsp><NRf>[DB]
ASE ALGO	

ANALYSIS Parameters	Control Command
AUTO-FIX	:CALCulate:PARameter[:CATegory]:NF:AALGo<wsp>AFIX|0
MANUAL-FIX	:CALCulate:PARameter[:CATegory]:NF:AALGo<wsp>MFIX|1
AUTO-CTR	:CALCulate:PARameter[:CATegory]:NF:AALGo<wsp>ACENter|2
MANUAL-CTR	:CALCulate:PARameter[:CATegory]:NF:AALGo<wsp>MCENter|3
FITTING AREA	:CALCulate:PARameter[:CATegory]:NF:FAREa<wsp><NRf>[M]
MASK AREA	:CALCulate:PARameter[:CATegory]:NF:MAREa<wsp><NRf>[M]
FITTING ALGO	
LINEAR	:CALCulate:PARameter[:CATegory]:NF:FALGo<wsp>LINear|0
GAUSS	:CALCulate:PARameter[:CATegory]:NF:FALGo<wsp>GAUSs|1
LORENZ	:CALCulate:PARameter[:CATegory]:NF:FALGo<wsp>LORenz|2
3RD POLY	:CALCulate:PARameter[:CATegory]:NF:FALGo<wsp>3RD|3
4TH POLY	:CALCulate:PARameter[:CATegory]:NF:FALGo<wsp>4TH|4
5TH POLY	:CALCulate:PARameter[:CATegory]:NF:FALGo<wsp>5TH|5
POINT DISPLAY ON/OFF	:CALCulate:PARameter[:CATegory]:NF:PDISplay<wsp> OFF|ON|0|1
NF CALCULATION SETTING	
RES BW	:CALCulate:PARameter[:CATegory]:NF:RBWidth<wsp> MEASured|CAL|0|1
SHOT NOISE	:CALCulate:PARameter[:CATegory]:NF:SNOise<wsp>OFF|ON|0|1
OTHER SETTING	
SIGNAL POWER	:CALCulate:PARameter[:CATegory]:NF:SPOWer<wsp>PEAK| INTegral|0|1
INTEGRAL RANGE	:CALCulate:PARameter[:CATegory]:NF:IRANge<wsp><NRf>
FILTER-PK	
PEAK LEVEL	
SW ON/OFF	:CALCulate:PARameter[:CATegory]:FILPk<wsp>PLEVel,SW, OFF|ON|0|1
PEAK WAVELENGTH	
SW ON/OFF	:CALCulate:PARameter[:CATegory]:FILPk<wsp>PWAVelength, SW,OFF|ON|0|1
CENTER WAVELENGTH	
SW ON/OFF	:CALCulate:PARameter[:CATegory]:FILPk<wsp>MWAVelength,SW, OFF|ON|0|1
ALGO	:CALCulate:PARameter[:CATegory]:FILPk<wsp>MWAVelength, ALGO,<data>
THRESH LEVEL **.**dB	:CALCulate:PARameter[:CATegory]:FILPk<wsp>MWAVelength, TH,<NRf>[DB]
K	:CALCulate:PARameter[:CATegory]:FILPk<wsp>MWAVelength,K,<NRf>
MODE FIT ON/OFF	:CALCulate:PARameter[:CATegory]:FILPk<wsp>MWAVelength, MFIT,OFF|ON|0|1
MODE DIFF *.**dB	:CALCulate:PARameter[:CATegory]: FILPk<wsp> MWAVelength, MDIFf,<NRf>[DB]
SPECTRUM WIDTH	
SW ON/OFF	:CALCulate:PARameter[:CATegory]:FILPk<wsp>SWIDth,SW, OFF|ON|0|1
ALGO	:CALCulate:PARameter[:CATegory]:FILPk<wsp> SWIDth, ALGO,<data>
THRESH LEVEL **.**dB	:CALCulate:PARameter[:CATegory]:FILPk<wsp> SWIDth, TH,<NRf>[DB]
K	:CALCulate:PARameter[:CATegory]: FILPk<wsp> SWIDth, K,<NRf>
MODE FIT ON/OFF	:CALCulate:PARameter[:CATegory]: FILPk<wsp> SWIDth, MFIT,OFF|ON|0|1
MODE DIFF *.**dB	:CALCulate:PARameter[:CATegory]: FILPk<wsp> SWIDth, MDIFf,<NRf>[DB]

ANALYSIS Parameters	Control Command
RIPPLE WIDTH	
SW ON/OFF	:CALCulate:PARameter[:CATegory]:FILPk<wsp>RWIDth,SW,OFF|ON|0|1
THRESH LEVEL **.**dB	:CALCulate:PARameter[:CATegory]:FILPk<wsp>RWIDth,TH,<NRf>[DB]
MODE DIFF *.**dB	:CALCulate:PARameter[:CATegory]:FILPk<wsp>RWIDth,MDIFf,<NRf>[DB]
CROSS TALK	
SW ON/OFF	:CALCulate:PARameter[:CATegory]:FILPk<wsp>XTALk,SW,OFF|ON|0|1
ALGO	:CALCulate:PARameter[:CATegory]:FILPk<wsp>XTALk,ALGO,<data>
THRESH LEVEL **.**dB	:CALCulate:PARameter[:CATegory]:FILPk<wsp>XTALk,TH,<NRf>[DB]
K	:CALCulate:PARameter[:CATegory]: FILPk<wsp>XTALk,K,<NRf>
MODE FIT ON/OFF	:CALCulate:PARameter[:CATegory]:FILPk<wsp>XTALk,MFIT,OFF|ON|0|1
MODE DIFF *.**dB	:CALCulate:PARameter[:CATegory]:FILPk<wsp>XTALk,MDIFf,<NRf>[DB]
CH SPACE ±*.**nm	:CALCulate:PARameter[:CATegory]:FILPk<wsp>XTALk,CSPace,<NRf>[M]
SEARCH AREA ±*.**nm	:CALCulate:PARameter[:CATegory]: FILPk<wsp>XTALk,SARea,<NRf>[M]
FILTER-BTM	
BOTTOM LEVEL	
SW ON/OFF	:CALCulate:PARameter[:CATegory]:FILBtm<wsp>BLEVel,SW,OFF|ON|0|1
BOTTOM WAVELENGTH	
SW ON/OFF	:CALCulate:PARameter[:CATegory]:FILBtm<wsp>BWAVelength,SW,OFF|ON|0|1
CENTER WAVELENGTH	
SW ON/OFF	:CALCulate:PARameter[:CATegory]:FILBtm<wsp>CWAVelength,SW,OFF|ON|0|1
ALGO	:CALCulate:PARameter[:CATegory]:FILBtm<wsp>CWAVelength,ALGO,<data>
THRESH LEVEL **.**dB	:CALCulate:PARameter[:CATegory]:FILBtm<wsp>CWAVelength,TH,<NRf>[DB]
MODE DIFF *.**dB	:CALCulate:PARameter[:CATegory]:FILBtm<wsp>CWAVelength,MDIFf,<NRf>[DB]
NOTCH WIDTH	
SW ON/OFF	:CALCulate:PARameter[:CATegory]:FILBtm<wsp>NWIDth,SW,OFF|ON|0|1
ALGO	:CALCulate:PARameter[:CATegory]:FILBtm<wsp>NWIDth,ALGO,<data>
THRESH LEVEL **.**dB	:CALCulate:PARameter[:CATegory]:FILBtm<wsp>NWIDth,TH,<NRf>[DB]
MODE DIFF *.**dB	:CALCulate:PARameter[:CATegory]:FILBtm<wsp>NWIDth,MDIFf,<NRf>[DB]
CROSS TALK	
SW ON/OFF	:CALCulate:PARameter[:CATegory]:FILBtm<wsp>XTALk,SW,OFF|ON|0|1
ALGO	:CALCulate:PARameter[:CATegory]:FILBtm<wsp>XTALk,ALGO,<data>
THRESH LEVEL **.**dB	:CALCulate:PARameter[:CATegory]:FILBtm<wsp>XTALk,TH,<NRf>[DB]
MODE DIFF *.**dB	:CALCulate:PARameter[:CATegory]:FILBtm<wsp>XTALk,MDIFf,<NRf>[DB]

ANALYSIS Parameters	Control Command
CH SPACE ±*.**nm	:CALCulate:PARameter[:CATegory]:FILBtm<wsp>XTALk, CSPace,<NRf>[M]
SEARCH AREA ±*.**nm	:CALCulate:PARameter[:CATegory]:FILBtm<wsp>XTALk,SAREa, <NRf>[M]
WDM FIL-PK	
CHANNEL DETECTION/NOMINAL WAVELENGTH	
ALGO	:CALCulate:PARameter[:CATegory]:WFPeak<wsp>NWAVelength, ALGO,<data>
THRESH LEVEL **.**dB	:CALCulate:PARameter[:CATegory]:WFPeak<wsp>NWAVelength,TH, <NRf>[DB]
MODE DIFF *.**dB	:CALCulate:PARameter[:CATegory]:WFPeak<wsp>NWAVelength, MDIFf,<NRf>[DB]
TEST BAND *.***nm	:CALCulate:PARameter[:CATegory]:WFPeak<wsp>NWAVelength, TBANd<NRf>[M]
PEAK WAVELENGTH/LEVEL	
SW ON/OFF	:CALCulate:PARameter[:CATegory]:WFPeak<wsp>PWAVelength,SW, OFF|ON|0|1
XdB WIDTH / CENTER WAVELENGTH	
SW ON/OFF	:CALCulate:PARameter[:CATegory]:WFPeak<wsp>CWAVelength,SW, OFF|ON|0|1
THRESH LEVEL **.**dB	:CALCulate:PARameter[:CATegory]:WFPeak<wsp>CWAVelength, TH,<NRf>[DB]
XdB STOP BAND	
SW ON/OFF	:CALCulate:PARameter[:CATegory]:WFPeak<wsp>SBANd,SW, OFF|ON|0|1
THRESH LEVEL **.**dB	:CALCulate:PARameter[:CATegory]:WFPeak<wsp>SBANd,TH, <NRf>[DB]
XdB PASS BAND	
SW ON/OFF	:CALCulate:PARameter[:CATegory]:WFPeak<wsp>PBANd,SW, OFF|ON|0|1
THRESH LEVEL	:CALCulate:PARameter[:CATegory]:WFPeak<wsp>PBANd,TH, <NRf>[DB]
TEST BAND *.***nm	:CALCulate:PARameter[:CATegory]:WFPeak<wsp>PBANd,TBANd, <NRf>[M]
RIPPLE	
SW ON/OFF	:CALCulate:PARameter[:CATegory]:WFPeak<wsp>RIPPle,SW, OFF|ON|0|1
TEST BAND *.***nm	:CALCulate:PARameter[:CATegory]:WFPeak<wsp>RIPPle,TBANd, <NRf>[M]
CROSS TALK	
SW ON/OFF	:CALCulate:PARameter[:CATegory]:WFPeak<wsp>XTALk,SW, OFF|ON|0|1
SPACING *.**nm	:CALCulate:PARameter[:CATegory]:WFPeak<wsp>XTALk,SPACing, <NRf>[M]
TEST BAND *.***nm	:CALCulate:PARameter[:CATegory]:WFPeak<wsp>XTALk,TBANd, <NRf>[M]
WDM FIL-BTM	
CHANNEL DETECTION/ NOMINAL WAVELENGTH	
ALGO	:CALCulate:PARameter[:CATegory]:WFBottom<wsp>NWAVelength, ALGO,<data>
THRESH LEVEL **.**dB	:CALCulate:PARameter[:CATegory]:WFBottom<wsp>NWAVelength, TH,<NRf>[DB]
MODE DIFF *.**dB	:CALCulate:PARameter[:CATegory]:WFBottom<wsp>NWAVelength, MDIFf,<NRf>[DB]
TEST BAND *.***nm	:CALCulate:PARameter[:CATegory]:WFBottom<wsp>NWAVelength, TBANd<NRf>[M]

ANALYSIS Parameters	Control Command
BOTTM WAVELENGTH/LEVEL	
SW ON/OFF	:CALCulate:PARameter[:CATegory]:WFBottom<wsp>BWAVelength,SW,OFF|ON|0|1
XdB NOTCH WIDTH/CENTER WAVELENGTH	
SW ON/OFF	:CALCulate:PARameter[:CATegory]:WFBottom<wsp>CWAVelength,SW,OFF|ON|0|1
ALGORITHM	:CALCulate:PARameter[:CATegory]:WFBottom<wsp>CWAVelength,ALGO,<data>
THRESH LEVEL **.**dB	:CALCulate:PARameter[:CATegory]:WFBottom<wsp>CWAVelength,TH,<NRf>[DB]
XdB STOP BAND	
SW ON/OFF	:CALCulate:PARameter[:CATegory]:WFBottom<wsp>SBANd,SW,OFF|ON|0|1
THRESH LEVEL **.**dB	:CALCulate:PARameter[:CATegory]:WFBottom<wsp>SBANd,TH,<NRf>[DB]
WDM FIL-BTM	
XdB ELIMINATION BAND	
SW ON/OFF	:CALCulate:PARameter[:CATegory]:WFBottom<wsp>EBANd,SW,OFF|ON|0|1
THRESH LEVEL **.**dB	:CALCulate:PARameter[:CATegory]:WFBottom<wsp> EBANd,TH,<NRf>[DB]
TEST BAND *.***nm	:CALCulate:PARameter[:CATegory]:WFBottom<wsp>EBANd,TBANd,<NRf>[M]
RIPPLE	
SW ON/OFF	:CALCulate:PARameter[:CATegory]:WFBottom<wsp>RIPPle,SW,OFF|ON|0|1
TEST BAND *.***nm	:CALCulate:PARameter[:CATegory]:WFBottom<wsp>RIPPle,TBANd,<NRf>[M]
CROSS TALK	
SW ON/OFF	:CALCulate:PARameter[:CATegory]:WFBottom <wsp>XTALk,SW,OFF|ON|0|1
SPACING *.**nm	:CALCulate:PARameter[:CATegory]:WFBottom<wsp>XTALk,SPACing,<NRf>[M]
TEST BAND *.***nm	:CALCulate:PARameter[:CATegory]:WFBottom<wsp>XTALk,TBANd,<NRf>[M]

Appendix 3 Output Format for Analysis Results

Output of Analysis Results

The analysis results of analysis functions are collectively output using the :CALCulate:DATA? command. If analysis has been not performed, a query error occurs.

Output Data Format for Each Analysis Function

The output data format of each analysis function is as shown below.
For information on abbreviations such as <center wl>, see “List of Abbreviations of Data Output using the :CALCulate:DATA? Command.”

THRESH , ENVELOPE, PK-RMS
<center wl>,<spec wd>,<mode num>

RMS
<center wl>,<spec wd>

NOTCH
<center wl>,<notch wd>

SMSR

- **SMSR1, SMSR2**
  <peak wl>,<peak lvl>,<2nd peak wl>,<2nd peak lvl>,<delta wl>,<delta lvl>
- **SMSR3, SMSR4**
  <peak wl>,<peak lvl>,<2nd peak wl(L)>,<2nd peak lvl(L)>,<delta wl(L)>,<delta lvl(L)>, <2nd peak wl(R)>,<2nd peak lvl(R)>,<delta wl(R)>,<delta lvl(R)>

POWER
<total pow>

DFB-LD
<spec wd>,<peak wl>,<peak lvl>,<mode ofst>,<smsr>

FP-LD
<spec wd>,<peak wl>,<peak lvl>,<center wl>,<total pow>,<mode num>

LED
<spec wd>,<peak wl>,<peak lvl>,<center wl>,<total pow>

WDM

- **ABSOLUTE, CH RELATION=OFFSET, OSNR DISPLAY=ON**
  <ch num>,<center wl>,<peak lvl>,<offset wl>,<offset lvl>,<noise>,<snr>,...

- **ABSOLUTE, CH RELATION=OFFSET, OSNR DISPLAY=OFF**
  <ch num>,<center wl>,<peak lvl>,<offset wl>,<offset lvl>,...

- **ABSOLUTE, CH RELATION=SPACING, OSNR DISPLAY=ON**
  <ch num>,<center wl>,<peak lvl>,<spacing>,<lvl diff>,<noise>,<snr>,...

- **ABSOLUTE, CH RELATION=SPACING, OSNR DISPLAY=OFF**
  <ch num>,<center wl>,<peak lvl>,<spacing>,<lvl diff>,...

- **RELATIVE, OSNR DISPLAY=ON**
  <ch num>,<grid wl>,<center wl>,<rel wl>,<peak lvl>,<noise>,<snr>,...

- **RELATIVE, OSNR DISPLAY=OFF**
  <ch num>,<grid wl>,<center wl>,<rel wl>,<peak lvl>,...

- **DRIFT (MEAS)**
  <ch num>,<ref wl>,<center wl>,<wl diff max>,<wl diff min>,<ref lvl>,<peak lvl>,<lvl diff max>,
  <lvl diff min>,...

- **DRIFT (GRID)**
  <ch num>,<grid wl>,<center wl>,<wl diff max>,<wl diff min>,<ref lvl>, <peak lvl>, <lvl diff max>,
  <lvl diff min>,...

EDFA-NF

<ch num>,<center wl>,<input lvl>,<output lvl>,<ase lvl>, <resoln>, <gain>, <nf>,...

FILTER-PK

<peak wl>,<peak lvl>,<center wl>,<spec wd>,<l-xtalk>,<r-xtalk>, <ripple>

FILTER-BTM

<btm wl>, <btm lvl>, <center wl>, <notch wd>, <l-xtalk>, <r-xtalk>

WDM FIL-PK

<ch num>, <nominal wl>, <peak wl>, <peak lvl>, <xdb wd>, <center wl>, <xdb sb>, <xdb pb>,
<ripple>, <l-xtalk>, <r-xtalk>,...
* Items with SW set to OFF are also output.

WDM FIL-BTM

<ch num>, <nominal wl>, <btm wl>, <btm lvl>, <xdb ntwd>, <center wl>, <xdb sb>, <xdb eb>,
<ripple>, <l-xtalk>, <r-xtalk>,...
* Items with SW set to OFF are also output.

WDM SMSR

<ch num>, <peak wl>, <peak lvl>, <spec wd>, <2nd peak wl>, <2nd peak lvl>, <smsr>, …

* Items with SW set to OFF are also output.

COLOR

<dominant wl>,<x col>,<y col>,<z col>,<color temp>,<dev>

TLS

<peak wl>,<peak lvl>,<center wl>,<spec wd>,<smsr(L)>,<smsr(R)>,<mode ofst(L)>, <mode ofst(R)>,<power>,<sser>,<stsser>

List of Abbreviations of Data Output Using the :CALCulate:DATA? Command

Abbreviation	Description	format	Output Unit
<center wl>	Center wavelength	<NRf>	m/Hz
<spec wd>	Spectrum width	<NRf>	m/Hz
<mode num>	Mode number	<integer>	
<notch wd>	Notch width	<NRf>	m/Hz
<peak wl>	Peak wavelength	<NRf>	m/Hz
<peak lvl>	Peak level	<NRf>	dBm
<2nd peak wl>	2nd peak wavelength	<NRf>	m/Hz
<2nd peak lvl>	2nd peak level	<NRf>	dB
<delta wl>	Wavelength difference	<NRf>	m/Hz
<delta lvl>	Level difference	<NRf>	dB
<mode ofst>	Mode offset	<NRf>	m/Hz
<smsr>	SMSR value	<NRf>	dB
<smsr(L)>	SMSR value (shorter wavelength side)	<NRf>	dB
<smsr(R)>	SMSR value (longer wavelength side)	<NRf>	dB
<mode ofst(L)>	Mode offset (shorter wavelength side)	<NRf>	m/Hz
<mode ofst(R)>	Mode offset (longer wavelength side)	<NRf>	m/Hz
<2nd peak wl(L)>	Second peak wavelength (shorter wavelength side)	<NRf>	m/Hz
<2nd peak wl(R)>	Second peak wavelength (longer wavelength side)	<NRf>	m/Hz
<2nd peak lvl(L)>	Second peak level (shorter wavelength side)	<NRf>	dB
<2nd peak lvl(R)>	Second peak level (longer wavelength side)	<NRf>	dB
<delta wl(L)>	Wavelength difference (shorter wavelength side)	<NRf>	m/Hz
<delta wl(R)>	Wavelength difference (longer wavelength side)	<NRf>	m/Hz
<delta lvl(L)>	Level difference (shorter wavelength side)	<NRf>	dB
<delta lvl(R)>	Level difference (longer wavelength side)	<NRf>	dB
<power>	Power value	<NRf>	dBm/W
<total pow>	Total power value	<NRf>	dBm/W
<mode num>	Mode number	<integer>	
<left mode peak>	Mode peak frequency (left)	<NRf>	Hz
<right mode peak>	Mode peak frequency (right)	<NRf>	Hz
<pmd>	PMD value	<NRf>	ps
<ch num>	Channel number	<integer>	
<offset wl>	Offset wavelength	<NRf>	m/Hz
<offset lvl>	Offset level	<NRf>	dB
<noise>	Noise level	<NRf>	dBm/NBW
<snr>	SNR value	<NRf>	dB
<grid wl>	Grid wavelength	<NRf>	m/Hz
<rel wl>	Relative wavelength	<NRf>	m/Hz
<wl diff max>	Wavelength difference (max.)	<NRf>	m/Hz
<wl diff min>	Wavelength difference (min.)	<NRf>	m/Hz

Abbreviation	Description	format	Output Unit
<ref lvl>	Relative level	<NRf>	dB
<lvl diff max>	Level difference (max.)	<NRf>	dB
<lvl diff min>	Level difference (min.)	<NRf>	dB
<input lvl>	Input level	<NRf>	dBm
<output lvl>	Output level	<NRf>	dBm
<ase lvl>	ASE level	<NRf>	dBm/RES
<resoln>	Measurement resolution	<NRf>	m
<gain>	Gain	<NRf>	dB
<nf>	NF value	<NRf>	dB
<l-xtalk>	Crosstalk value (left)	<NRf>	dB
<r-xtalk>	Crosstalk value (right)	<NRf>	dB
<ripple>	Ripple width	<NRf>	m/Hz
<nominal wl>	Reference wavelength	<NRf>	m/Hz
<xdb wd>	Xdb width	<NRf>	m/Hz
<xdb sb>	XdB stop-band	<NRf>	m/Hz
<xdb pb>	XdB pass-band	<NRf>	m/Hz
<xdb eb>	XdB elimination-band	<NRf>	m/Hz
<dominant wl>	Dominant Wavelength	<NRf>	m/Hz
<x col>	Chromaticity coordinates (x)	<NRf>	
<y col>	Chromaticity coordinates (y)	<NRf>	
<z col>	Chromaticity coordinates (z)	<NRf>	
<color temp>	Color temperature	<NRf>	K
<dev>	Deviation	<NRf>	

Appendix 4 List of the AQ6317-Compatible Commands

For compatibility with the AQ6374E, see the following table, AQ6317-Compatible Commands.

AQ6317 Series Control Command	Operates in AQ6317 Comp Mode	AQ6374E Control Command Corresponding to AQ6317 Command	Remarks
3D	×	-	
3DRCL	×	-	
A+BCL	○	:CALCulate:MATH:TRC <wsp>A+B(LIN)	
A=B	○	:TRACe:COPY<wsp>TRB,TRA	
A=C	○	:TRACe:COPY<wsp>TRC,TRA	
A-BC	○	:CALCulate:MATH:TRC <wsp>A-B(LOG)	
A-BCL	○	:CALCulate:MATH:TRC <wsp>A-B(LIN)	
ACTV*	○	:TRACe:ACTive<wsp> <trace name> <trace name> = TRA|TRB|TRC	
ANA?	○	:CALCulate:DATA?	Diff. talker format
ANGL***	×	-	
AREA*	×	-	
ARES?	×	-	
ARESDSP*	×	-	
ATANA*	○	:CALCulate[:IMMediate]: AUTO<wsp>OFF|ON|0|1	
ATCTR*	○	:CALCulate:MARKer: MAXimum:SCENter: AUTO<wsp>OFF|ON|0|1	
ATOFS*	○	:CALibration:ZERO[:AUTO] <wsp>OFF|ON|0|1|	
ATREF*	○	:CALCulate:MARKer: MAXimum:SRLevel:AUTO	
ATSCL*	○	:DISPlay[:WINDow]:TRACe: Y2[:SCALe]:AUTO<wsp> OFF|ON|0|1	
ATSR*	○	:CALCulate:MARKer:AUTO <wsp>OFF|ON|0|1	
AUTO	○	:INITIate:SMODe<wsp> AUTO|3;INITiate	
AVG****	▲	:SENSe:AVERage:COUNt <wsp><integer>	Diff. parameter range
B=A	○	:TRACe:COPY<wsp>TRA,TRB	
B=C	○	:TRACe:COPY<wsp>TRC,TRB	
B-AC	○	:CALCulate:MATH:TRC <wsp>B-A(LOG)	
B-ACL	○	:CALCulate:MATH:TRC <wsp>B-A(LIN)	
BASL***.*	○	:DISPlay[:WINDow]:TRACe: Y1[:SCALe]:SPACing<wsp> LINear|1; :DISPlay[:WINDow]:TRACe: Y1[:SCALe]:BLEVel<wsp> <NRf>[MW]	
BD*	○	-	
BLKA	○	:TRACe:STATe:TRA <wsp>OFF|0	

App

Appendix

AQ6317 Series Control Command	Operates in AQ6317 Comp Mode	AQ6374E Control Command Corresponding to AQ6317 Command	Remarks
BLKB	○	:TRACe:STATe:TRB <wsp>OFF|0	
BLKC	○	:TRACe:STATe:TRC <wsp>OFF|0	
BTSR	○	:CALCulate:MARKer: MINimum	
BZCLK*	○	:SYSTem:BUZZer:CLICk <wsp>OFF|ON|0|1	
BZWRN*	○	:SYSTem:BUZZer:WARNing <wsp>OFF|ON|0|1	
C=A	○	:TRACe:COPY<wsp>TRA,TRC	
C=B	○	:TRACe:COPY<wsp>TRC,TRB	
CLMES	○	-	
CLR	○	:TRACe:DELete<wsp>TRA; :TRACe:DELete<wsp>TRB; :TRACe:DELete<wsp>TRC	
CNDDT*	○	:MMEMory:STORe:DATA:ITEM <wsp>CONDition,OFF|ON|0|1	
COPY*	×	-	
CRS*	○	-	
CTR=M	○	:CALCulate:MARKer:SCENter	
CTR=P	○	:CALCulate:MARKer:MAXimum: SCENter	
CTRF***.***	▲	:SENSe:WAVelength:CENTer <wsp><NRf>[HZ]	Diff.parameter range
CTRWL****.**	▲	:SENSe:WAVelength:CENTer <wsp><NRf>[M]	Diff.parameter range
CVFTC**	×	-	Same cmd for TRACE G
CVPKC**	×	-	Same cmd for TRACE G
CWPLS?	▲	-	Diff. query data 0:Except CW 1:CW
D&TDT*	○	:MMEMory:STORe:DATA:ITEM <wsp>DATE,OFF|ON|0|1	
DATE?	○	:SYSTem:DATE?	Diff. talker format
DATE YR.MO.DY	○	:SYSTem:DATE<wsp> <year>,<month>,<day>	
TIME HH:MM	○	:SYSTem:TIME<wsp> <hour>,<minute>,<second>	
DEFCL*	▲	:DISPlay:COLor<wsp> <mode>	Diff. display color <mode>=0:B&W, 1=color mode
DEL'@@@@.***'	○	:MMEMory:DELete<wsp> <"file name">,EXTernal	
DFBAN	○	:CALCulate:CATegory<wsp> DFBLd|6	
DFBLD○;□;▲;****	▲	-	
DIR?	×	-	
DISP?	○	-	
DSPA	○	:TRACe:STATe:TRA<wsp> ON|1	
DSPB	○	:TRACe:STATe:TRB<wsp> ON|1	
DSPA?	○	:TRACe:STATe:TRA?	
DSPB?	○	:TRACe:STATe:TRB?	
DSPC	○	:TRACe:STATe:TRC<wsp> ON|1	

AQ6317 Series Control Command	Operates in AQ6317 Comp Mode	AQ6374E Control Command Corresponding to AQ6317 Command	Remarks
DSPC?	○	:TRACe:STATe:TRC?	
DTAD*	○	:MMEMory:STORe:DATA:MODE<wsp>ADD|OVER|0|1	
DTARA*	○	:MMEMory:STORe:DATA:ITEM<wsp>DATA,OFF|ON|0|1	
DUTCH***;####.##	×	-	
DUTCHF***;###.###	×	-	
DUTLEV**.**	×	-	
DUTSNR**.**	×	-	
EDFCVF*	×	-	
EDFTH**.*	×	-	
EDNF	×	-	
ENVK**.**	○	:CALCulate:PARameter[:CATegory]:SWENvelope:K<wsp><NRf>	
ENVT1**.**	○	:CALCulate:PARameter[:CATegory]:SWENvelope:TH1<wsp><NRf>[DB]	
ENVT2**.**	○	:CALCulate:PARameter[:CATegory]:SWENvelope:TH2<wsp><NRf>[DB]	
EXEC**	×	-	
EXTRG	○	:TRIGger[:SEQuence]:STATe<wsp>OFF|ON|0|1	
FIG*	○	:UNIT:POWer:DIGit<wsp>1|2|3	
FILBTM○;□;▲;***	▲	:CALCulate:PARameter[:CATegory]:FILBtm<wsp><item>,<paramater>,<data>	
FILBTMAN	○	:CALCulate:CATegory<wsp>FILBtm|14	
FILPK○;□;▲;***	▲	:CALCulate:PARameter[:CATegory]:FILPk<wsp><item>,<paramater>,<data>	
FILPKAN	○	:CALCulate:CATegory<wsp>FILPk|13	
FIXA	○	:TRACe:ATTRibute:TRA<wsp>FIX|1	
FIXB	○	:TRACe:ATTRibute:TRB<wsp>FIX|1	
FIXC	○	:TRACe:ATTRibute:TRC<wsp>FIX|1	
FMKR***.****	▲	:CALCulate:MARKer:X<wsp>0,<NRf>[HZ]	Diff. parameter range
FPAN	○	:CALCulate:CATegory<wsp>FPLD|6	
FPLD;○;□;▲;****	▲	:CALCulate:PARameter[:CATegory]:FPLD<wsp><item>,<paramemter>,<data>	
GP2ADR**	×	-	
GRCOL*	▲	-	Valid only when the parameter is 0 or 1
GRFMT*	○	-	
HD*	○	-	
HELP*	×	-	
*IDN?	○	*IDN?	
INIT	○	:SYSTem:PRESet	

App

Appendix

AQ6317 Series Control Command	Operates in AQ6317 Comp Mode	AQ6374E Control Command Corresponding to AQ6317 Command	Remarks
KABC	○	:CALCulate:MATH:TRC <wsp>1-K(A/B)	
KABCK*****.****	○	:CALCulate:MATH:TRC: K<wsp><NRf>	
KBAC	○	:CALCulate:MATH:TRC <wsp>1-K(B/A)	
KYDNE	×	–	
L1FMK***.****	▲	:CALCulate:LMARker: X<wsp>1,<NRf>[HZ]	Diff. parameter range
L1MK****.*	▲	:CALCulate:LMARker: X<wsp>1,<NRf>[M]	Diff. parameter range
L1MK?	▲	:CALCulate:LMARker: X?<wsp>1	Diff. parameter range
L2FMK****.****	▲	:CALCulate:LMARker: X<wsp>2,<NRf>[HZ]	Diff. parameter range
L2MK****.***	▲	:CALCulate:LMARker: X<wsp>2,<NRf>[M]	Diff. parameter range
L2MK?	▲	:CALCulate:LMARker: X?<wsp>2	Diff. parameter range
L3DB****.**	▲	:CALCulate:LMARker: Y<wsp>3,<NRf>[DB]	Diff. parameter range
L3DBM****.**	▲	:CALCulate:LMARker: Y<wsp>3,<NRf>[DBM]	Diff. parameter range
L3LN*.***E±**	▲	:CALCulate:LMARker: Y<wsp>3,<NRf>	Diff. parameter range
L3MK?	▲	:CALCulate:LMARker: Y?<wsp>3	Diff. parameter range
L4DB****.**	▲	:CALCulate:LMARker: Y<wsp>4,<NRf>[DB]	Diff. parameter range
L4DBM****.**	▲	:CALCulate:LMARker: Y<wsp>4,<NRf>[DBM]	Diff. parameter range
L4LN*.***E±**	▲	:CALCulate:LMARker: Y<wsp>4,<NRf>	Diff. parameter range
L4MK?	▲	:CALCulate:LMARker: Y?<wsp>4	Diff. parameter range
LBL '*********'	▲	:DISPlay[:WINDow]:TEXT: DATA<wsp><string>	Diff. no. of chars
LBLCL	○	:DISPlay[:WINDow]:TEXT: CLEar	
LBLDT*	○	:MMEMory:STORe:DATA: ITEM<wsp>LABel, OFF|ON|0|1	
LCALT****;#.###	×	-	
LDATAR****-R****	○	:TRACe[:DATA]:X?<wsp> <trace name> [,<start point>, <stop point>]	
LDATBR****-R****	○	:TRACe[:DATA]:Y?<wsp> <trace name> [,<start point>, <stop point>]	
LDATCR****-R****	○	:TRACe[:DATA]:SNUMber? <wsp><trace name>	
WDATAR****-R****	○		
WDATBR****-R****	○		
WDATCR****-R****	○		
DTNUM A	○		

AQ6317 Series Control Command	Operates in AQ6317 Comp Mode	AQ6374E Control Command Corresponding to AQ6317 Command	Remarks
DTNUM B	○		
DTNUM C	○		
LMEM$$R****-R****	×	-	
WMEM$$R****-R****	×	-	
DTNUM **	×	-	
LDTDIG*	○	-	
LED○;□;▲;*****	▲	:CALCulate:PARameter [:CATegory]:LED<wsp> <item>,<paramater>,<data>	
LEDAN	○	:CALCulate:CATegory <wsp>LED|7	
LHLD*	×	-	
LMKCL	○	:CALCulate:LMARker:AOFF	
LNGT**.***	○	:DISPlay[:WINDow]:TRACe: Y2[:SCALe]:LENGth	
LOFSKM***.*	○	:DISPlay[:WINDow]:TRACe: Y2[:SCALe]:OLEVel<wsp> <NRf>[DB/KM]	
LOFST***.*	○	:DISPlay[:WINDow]:TRACe: Y2[:SCALe]:OLEVel<wsp> <NRf>[DB]	
LOGLMT***	×	-	
LPF	×	-	
LSCL**.*	○	:DISPlay[:WINDow]:TRACe: Y1[:SCALe]:SPACing<wsp> LOGarithmic|0; :DISPlay[:WINDow]:TRACe: Y1[:SCALe]:PDIVision<wsp> <integer>[DIV]	
LSUNT*	○	:DISPlay[:WINDow]:TRACe:Y1 [:SCALe]:UNIT<wsp>DBM|DBM/ NM	
LTABS	×	-	
LTALM?	×	-	
LTALMDT?	×	-	
LTATSCL*	×	-	
LTATSET	×	-	
LTCH***	×	-	
LTCHCUR***	×	-	
LTINTVL****.*	×	-	
LTL	×	-	
LTLHI***.**	×	-	
LTLLOW***.**	×	-	
LTLVLCTR***.**	×	-	
LTLVLSCL**.*	×	-	
LTREFINI	×	-	
LTREFSET	×	-	
LTREL	×	-	
LTSNR	×	-	
LTSNRCTR***.**	×	-	
LTSNRLIM**.**	×	-	
LTSNRSCL**.*	×	-	
LTSWP	×	-	
LTTIME****	×	-	
LTTMCUR****	×	-	
LTWL	×	-	

AQ6317 Series Control Command	Operates in AQ6317 Comp Mode	AQ6374E Control Command Corresponding to AQ6317 Command	Remarks
LTWLCTR****.**	×	-	
LTWLLIM**.**	×	-	
LTWLSPN****.*	×	-	
LVSFT***.**	○	:SENSe:CORRection:LEVel:SHIFt<wsp><NRf>[DB]	
MAXA	○	:TRACe:ATTRibute:TRA<wsp>MAX|2	
MCLR***	▲	:CALCulate:MARKer[:STATe]<wsp><marker>,OFF|0	Diff. parameter range
MEM*	×	–	
MESWL*	○	:SENSe:CORRection:RVELocity:MEDium<wsp>AIR|VACuum|0|1	
MIMSK**.**	×	–	
MINB	○	:TRACe:ATTRibute:TRB<wsp>MIN|3	
MKCL	○	:CALCulate:MARKer:AOFF	
MKR***	▲	:CALCulate:MARKer[:STATe]<wsp><marker>,|ON|1	Diff. parameter range
MKR?	○	:CALCulate:MARKer:X?<wsp>0	
MKR?****	○	:CALCulate:MARKer:X?<wsp><marker>	Diff. parameter range
MKR1	○	:CALCulate:MARKer[:STATe]<wsp>1,|ON|1	
MKR1?	○	:CALCulate:MARKer:X?<wsp>1	
MKR2	○	:CALCulate:MARKer[:STATe]<wsp>2,|ON|1	
MKR2?	○	:CALCulate:MARKer:X?<wsp>2	
MKROS*	○	:CALCulate:MARKer:FUNCtion:FORMat<wsp>OFFSet|SPACing|0|1	
MKRPRT	×	-	
MKRUP*	○	:CALCulate:MARKer:FUNCtion:UPDate<wsp>OFF|ON|0|1	
MKUNT*	○	:CALCulate:MARKer:UNIT<wsp>WAVelength|FREQuency|0|1	
MLTMKR*	×	–	
MODFT*	○	:CALCulate:PARameter[:CATegory]:SWTHresh:MFIT<wsp>OFF|ON|0|1	
MODIF**.**	○	:CALCulate:PARameter:COMMon:MDIFf<wsp><NRf>[DB]	
MSKL*	○	:DISPlay[:WINDow]:TRACe:Y:NMASk:TYPE<wsp>VERTical|HORIzontal|0|1	
NCHMOD*	○	:CALCulate:PARameter[:CATegory]:NOTCh:TYPE<wsp>PEAK|BOTTom|0|1	

AQ6317 Series Control Command	Operates in AQ6317 Comp Mode	AQ6374E Control Command Corresponding to AQ6317 Command	Remarks
NCHTH**.*	○	:CALCulate:PARameter [:CATegory]:NOTCh:TH <wsp><NRf>[DB]	
NMSK****	▲	:DISPlay[:WINDow]:TRACe:Y:NMASk<wsp><NRf>[DB]	Diff. parameter range
NORMC	×	-	Same cmd for TRACE G
NORMD	×	-	
NSR	○	:CALCulate:MARKer:MAXimum:NEXT or :CALCulate:MARKer:MINimum:NEXT	
NSRL	○	:CALCulate:MARKer:MAXimum:LEFT or :CALCulate:MARKer:MINimum:LEFT	
NSRR	○	:CALCulate:MARKer:MAXimum:RIGHt or :CALCulate:MARKer:MINimum:RIGHt	
OFIN***.**	×	-	
OFOUT***.**	×	-	
OPALIGN	○	:CALibration:ALIGn [:IMMediate]	
PKHLD****	○	-	
PKSR	○	:CALCulate:MARKer:MAXimum	
PKSR?	○	-	
PLMES	×	-	•PKHLD**** when PEAK HOLD MODE •EXTRG when EXT TRIGGER MODE
PLMOD?	○	-	
PLMSK**.**	×	-	
PMD	×	-	
PMDTH**.**	×	-	
PMRPT	×	-	
PMRST	×	-	
PMSGL	×	-	
PMSTP	×	-	
PMST?	×	-	
PMUNT*	×	-	
POFS**.**	○	:CALCulate:PARameter [:CATegory]:POWer:OFFSet <wsp><NRf>[DB]	
PRDEL**	×	-	
PREXT	×	-	
PRFED**	×	-	
PRMK**.**	○	:CALCulate:PARameter [:CATegory]:SWPKrms:K<wsp><NRf>	
PRMTH**.*	○	:CALCulate:PARameter [:CATegory]:SWPKrms:TH<wsp><NRf>[DB]	
PWR	○	:CALCulate:CATegory <wsp>POWer|9	
RAVA***	○	:TRACe:ATTRibute:RAVG[:TRA]<wsp><integer>	

AQ6317 Series Control Command	Operates in AQ6317 Comp Mode	AQ6374E Control Command Corresponding to AQ6317 Command	Remarks
RAVB***	○	:TRACe:ATTRibute: RAVG:TRB<wsp><integer>	
RCLA**	×	-	
RCLB**	×	-	
RCLC**	×	-	
RD*'@@@@'	○	:MMEMory:LOAD:TRACe<wsp> <trace name>,<"file name">, EXTernal <trace name> = TRA|TRB|TRC	Loads external memory
RD3D*'@@@@'	×	-	
RDDT'@@@@'	×	-	
RDLT'@@@@'	×	-	
RDMEM**'@@@@'	×	-	
RDPRG**'@@@@'	×	-	
RDSET'@@@@'	○	:MMEMory:LOAD:SETTing<wsp> <"file name">,EXTernal	Loads external memory
RDTMP'@@@@'	×	-	
REF=M	○	:CALCulate:MARKer:SRLevel	
REF=P	○	:CALCulate:MARKer:MAXimum: SRLevel	
REFL***.*	▲	:DISPlay[:WINDow]:TRACe:Y1 [:SCALe]:SPACing<wsp> LOGarithmic|0; :DISPlay:[:WINDow]:TRACeY1 [:SCAle]:RLEVel<wsp><NRf> [DBM]	Diff. parameter range
REFLM*.**	○	:DISPlay[:WINDow]:TRACe:Y1 [:SCALe]:SPACing<wsp> LINear|1; :DISPlay[:WINDow]:TRACe:Y1 [:SCALe]:RLEVel<wsp><NRf>MW	
REFLN*.**	○	:DISPlay[:WINDow]:TRACe:Y1 [:SCALe]:SPACing<wsp> LINear|1; :DISPlay[:WINDow]:TRACe:Y1 [:SCALe]:RLEVel<wsp><NRf>NW	
REFLP*.**	×	-	
REFLU*.**	○	:DISPlay[:WINDow]:TRACe:Y1 [:SCALe]: SPACing<wsp>LINear|1; :DISPlay[:WINDow]:TRACe:Y1 [:SCALe]:RLEVel<wsp><NRf>UW	
REFL?	▲	:DISPlay:[:WINDow]:Y1 [:SCAle]:RLEVel?	Diff. parameter range
REL*	×	-	
RESCOR*	○	:SENSe:SETTing:CORRection <wsp>OFF|ON|0|1	
RESLN*.**	▲	:SENSe:BANDwidth|:BWIDth [:RESolution]<wsp><NRf>[M]	Diff. parameter range
RESLNF***	×	-	
RMSK**.**	○	:CALCulate:PARameter [:CATegory]:RMS:K<wsp> <NRf>	
RMSTH**.*	○	:CALCulate:PARameter [:CATegory]:RMS:TH<wsp> <NRf>[DB]	

AQ6317 Series Control Command	Operates in AQ6317 Comp Mode	AQ6374E Control Command Corresponding to AQ6317 Command	Remarks
RPT	○	:INITiate:SMODe<wsp> REPeat|2; :INITiate	
*RST	▲	*RST	Diff. operation
SAVEA**	×	-	
SAVEB**	×	-	
SAVEC**	×	-	
SENS?	○	:SENSe:SENSe?	
SD*	○	-	
SEGP****	×	-	
SGL	○	:INITIate:SMODe<wsp> SINGle|1;:INITiate	
SHI1	▲	:SENSe:SENSe<wsp>HIGH1|3	Chopper Unused
SHI2	▲	:SENSe:SENSe<wsp>HIGH2|4	Chopper Unused
SHI3	▲	:SENSe:SENSe<wsp>HIGH3|5	Chopper Unused
SKM**.*	○	:DISPlay[:WINDow]:TRACe: Y2[:SCALe]:UNIT<wsp>DB/KM|2 :DISPlay[:WINDow]:TRACe: Y2[:SCALe]:PDIVision<wsp> <NRf>[DB/KM]	
SLIN*.***	○	:DISPlay[:WINDow]:TRACe: Y2[:SCALe]:UNIT<wsp> LINear|1 :DISPlay[:WINDow]:TRACe: Y2[:SCALe]:PDIVision<wsp> <NRf>	
SLOG**.*	○	:DISPlay[:WINDow]:TRACe: Y2[:SCALe]:UNIT<wsp>DB|0 :DISPlay[:WINDow]:TRACe: Y2[:SCALe]:PDIVision<wsp> <NRf>[DB]	
SMEAS	×	-	
SMID	○	:SENSe:SENSe<wsp>MID|2	
SMIN***.*	○	:DISPlay[:WINDow]:TRACe: Y2[:SCALe]:SMINimum<wsp> <NRf>	
SMINP***.*	○	:DISPlay[:WINDow]:TRACe: Y2[:SCALe]:SMINimum<wsp> <NRf>[%]	
SMPL****	▲	:SENSe:SWEep:POINts<wsp> <integer>	Diff.parameter range
SMSR*	○	:CALCulate:PARameter [:CATegory]:SMSR:MODE <wsp>SMSR1|SMSR2	
SNAT	○	:SENSe:SENSe<wsp>NAUT|1	
SNHD	○	:SENSe:SENSe<wsp> NHLD|0	
SP=LM	○	:CALCulate:LMARker: SSPan	
SPAN****.*	▲	:SENSe:WAVelength: SPAN<wsp><NRf>[M]	Diff.parameter range
SPANF***.***	▲	-	Diff.parameter range
SPLIT	×	-	
SPN=W	○	-	

App

Appendix

AQ6317 Series Control Command	Operates in AQ6317 Comp Mode	AQ6374E Control Command Corresponding to AQ6317 Command	Remarks
SPS***.*	○	:DISPlay[:WINDow]:TRACe:Y2[:SCALe]:UNIT<wsp>%|3 :DISPlay[:WINDow]:TRACe:Y2[:SCALe]:PDIVision<wsp><NRf>[%]	
SRLMK*	○	:CALCulate:LMARker:SRANge<wsp>OFF|ON|0|1	
SRMSK***	○	-	
SRQ*	○	*SRE<wsp><integer>	
SSE*	×	-	
SSMSK**.**	○	:CALCulate:PARameter[:CATegory]:SMSR:MASK<wsp><NRf>[M]	
SSUNT?	○	:DISPlay[:WINDow]:TRACe:Y2[:SCALe]:UNIT?	
STAF***.***	▲	:SENSe:WAVelength:STARt<wsp><NRf>[HZ]	Diff. parameter range
STAWL****.**	▲	:SENSe:WAVelength:STARt<wsp><NRf>[M]	Diff. parameter range
STP	○	:ABORt	
STPF***.***	▲	:SENSe:WAVelength:STOP<wsp><NRf>[HZ]	Diff. parameter range
STPWL****.**	▲	:SENSe:WAVelength:STOP<wsp><NRf>[M]	Diff. parameter range
SW*	○	:CALCulate:CATegory<wsp>SWTHresh|0	
SWDSP*	×	-	
SWENV**.**	○	:CALCulate:PARameter[:CATegory]:SWENvelope:TH1<wsp><NRf>[DB]	
SWEEP?	○	-	
SWPI*****	○	:SENSe:SWEep:TIME:INTerval<wsp><integer>[SEC]	
SWPM*	○	:SENSe:WAVelength:SRANge<wsp>OFF|ON|0|1	
SWPRM**.**	○	:CALCulate:PARameter[:CATegory]:SWPKrms:TH<wsp><NRf>[DB]	
SWRMS**.**	○	:CALCulate:PARameter[:CATegory]:RMS:TH<wsp><NRf>[DB]	
SWTHR**.**	○	:CALCulate:PARameter[:CATegory]:SWTHresh:TH<wsp><NRf>[DB]	
THRK**.**	○	:CALCulate:PARameter[:CATegory]:SWTHresh:K<wsp><NRf>	
THRTH**.**	○	:CALCulate:PARameter[:CATegory]:SWTHresh:TH<wsp><NRf>[DB]	
TIME?	○	-	
TLDAT*****.**,***.**,***.**	×	-	
TLDATCLR	×	-	
TLDISP*	×	-	
TLEXTRA*	×	-	
TLGONO*	×	-	

AQ6317 Series Control Command	Operates in AQ6317 Comp Mode	AQ6374E Control Command Corresponding to AQ6317 Command	Remarks
TLSADR**	×	-	
TLSSYNC*	×	-	
TLLVSFT***.**	×	-	
TLRESLT?	×	-	
TLTYPE*	×	-	
TLWLSFT****.**	×	-	
TRA?	▲	:TRACe:ATTRibute:TRA?	Diff.talker format 2: MAX HOLD / MIN HOLD
TRB?	▲	:TRACe:ATTRibute:TRB?	Diff.talker format 2: MAX HOLD / MIN HOLD
TRC?	▲	:TRACe:ATTRibute:TRC?	
TRFMT*	○	-	
UCWRN*	○	:SYSTem:DISPlay:UNCal <wsp>OFF|ON|0|1	
UHLD*	×	-	
ULTRA*	×	-	
ULTRB*	×	-	
ULTRC*	×	-	
WARN?	▲	:SYSTem:ERRor[:NEXT]?	
WCAL****.***	▲	:CALibration:WAVelength:EXTernal:SOURce<wsp>LASer|0; :CALibration:WAVelength:EXTernal:WAVelength<wsp><NRf>[M]	Diff. parameter range
WCALG****.***	▲	:CALibration:WAVelength:EXTernal:SOURce<wsp>GASCell|1; :CALibration:WAVelength:EXTernal:WAVelength<wsp><NRf>[M]	Diff. parameter range
WCALS	○	:CALibration:WAVelength:INTernal[:IMMediate]	
WCALT****;#.###	×	-	
WDMAN	○	:CALCulate:CATegory <wsp>WDM|11	
WDMCHAUT*	×	-	No parameter
WDMCHSW***;#	×	-	
WDMDIF**.**	▲	:CALCulate:PARameter[:CATegory]:WDM:MDIFf <wsp><NRf>[DB]	Set only in WDM Analysis, not in NF Analysis
WDMDISP*	▲	:CALCulate:PARameter[:CATegory]:WDM:DTYPe <wsp><display type>	Diff. parameter 0: ABSOLUTE <display type>= ABSolute| 1: RELATIVE 0,RELatibe|1, MDRift|2,GDRift|3 3: DRIFT(MEAS) 4: DRIFT(GRID)
WDMDSPMSK***	▲	:CALCulate:PARameter[:CATegory]:WDM:DMASk <wsp><NRf>[DB]	Diff. parameter range
WDMDUAL*	○	:CALCulate:PARameter[:CATegory]:WDM:DUAL <wsp>OFF|ON|0|1	
WDMMAX***	×	-	No parameter
WDMMR	○	:CALCulate:PARameter[:CATegory]:WDM:MMReset	

App

Appendix

AQ6317 Series Control Command	Operates in AQ6317 Comp Mode	AQ6374E Control Command Corresponding to AQ6317 Command	Remarks
WDMNOI*	▲	[NOISE_ALGO is Auto Center] (NOISE POI=CTR) :CALCulate:PARameter [:CATegory] :WDM:NALGo<wsp>ACENter|2 [NOISE_ALGO is MANUAL Fix] (NOISE POI=CTR) :CALCulate:PARameter [:CATegory] :WDM:NALGo<wsp>MFIX|1; :CALCulate:PARameter [:CATegory] :WDM:FALGo<wsp>LINear|0;	Diff. set value 0: AUTO-FIX 1: AUTO-CTR Set only in WDM Analysis, not in NF Analysis
WDMNOIBW****	○	:CALCulate:PARameter [:CATegory]:WDM:NBW <wsp><NRf>[M|HZ]	
WDMNOIP**.**	▲	:CALCulate:PARameter [:CATegory]:	Valid only when NOISE ALGO is set to MANUAL FIX
WDMOS*	○	:CALCulate:PARameter [:CATegory]:WDM:RELation <wsp>OFFSet|SPACing|0|1	
WDMREF*	×	-	
WDMREFDAT*	×	-	
WDMRH	○	:CALCulate:PARameter [:CATegory]:WDM:RCH <wsp>0	
WDMRN***	○	:CALCulate:PARameter [:CATegory]:WDM:RCH <wsp><integer>	
WDMSLOPE*	○	:CALCulate:PARameter [:CATegory]:WDM:OSLope <wsp>OFF|ON|0|1	
WDMTCOPY	×	-	
WDMTH**.*	▲	:CALCulate:PARameter [:CATegory]:WDM:TH <wsp><NRf>[DB]	Set only in WDM Analysis, not in NF Analysis
WDMUNT*	×	:CALCulate:MARKer: UNIT<wsp>WAVelength| FREQuency|0|1	
WLSFT**.***	○	:SENSe:CORRection: WAVelength:SHIFt <wsp><NRf>[M]	
WMKR****.***	▲	:CALCulate:MARKer:X <wsp>0,<NRf>[M]	Diff. parameter range
WNFAN	○	:CALCulate:CATegory <wsp>NF|12	
WNFCVF*	▲	:CALCulate:PARameter [:CATegory]:NF:FALGo <wsp><algorithm> <algorithm>= AFIX|0,MFIX|1,ACENter|2 ,MCENter|3	Valid only when ASE ALGO is set to MANUAL FIX or MANUAL CTR
WNFFA**.**	▲	:CALCulate:PARameter [:CATegory]:NF:FARea <wsp><NRf>[M|HZ]	Valid only when ASE ALGO is set to MANUAL FIX or MANUAL CTR

AQ6317 Series Control Command	Operates in AQ6317 Comp Mode	AQ6374E Control Command Corresponding to AQ6317 Command	Remarks
WNFNP**.**	▲	:CALCulate:PARameter [:CATegory]:NF:MAREa <wsp><NRf>[M|HZ]	Valid only when all the following conditions are satisfied 1. ASE ALGO is set to MANUAL FIX or MANUAL CTR 2. FITTING ALGO is set besides LINEAR
WNFOFI***.**	○	:CALCulate:PARameter [:CATegory]:NF:IOFFset <wsp><NRf>[DB]	
WNFOFO***.**	○	:CALCulate:PARameter [:CATegory]:NF:OOFFset <wsp><NRf>[DB]	
WNFSSE*	×	-	No parameter
WR*'@@@@'	○	:MMEMory:STORe:TRACe <wsp><trace name>, BIN|CSV,<"file name">, EXTernal <trace name>=TRA|TRB|TRC	Saving to the external memory
WR3D*'@@@@'	×	-	
WRDT '@@@@'	○	:MMEMory:STORe:DATA <wsp><"file name">, EXTernal	Saving to the external memory
WRGR'@@@@'	○	:MMEMory:STORe:GRAPhics <wsp>B&W|COLor, BMP|JPG|PNG,<"file name">, EXTernal	Saving to the external memory
WRMEM**'@@@@'	×	-	
WRPRG**'@@@@'	×	-	
WRSET'@@@@'	○	:MMEMory:STORe:SETTing <wsp><"file name">, EXTernal	Saving to the external memory
WRTA	○	:TRACe:ATTRibute:TRA <wsp>WRITe|0	
WRTB	○	:TRACe:ATTRibute:TRB <wsp>WRITe|0	
WRTC	○	:TRACe:ATTRibute:TRC <wsp>WRITe|0	
WRTLT'@@@@'	×	-	
XUNT*	○	:UNIT:X<wsp>WAVelength| FREQuency|0|1	
ZSCL**	×	-	
ZSWPT**	○	:SENSe:SWEep:TIME:0NM<wsp> <integer>[SEC]	

App

Appendix

Appendix 5 HIGH1, HIGH2, HIGH3 of Measurement Sensitivity

Even when the measurement sensitivity of the instrument is set to HIGH1 or HIGH2 or HIGH3, the chopper cannot operate unless the CHOP MODE setting of the SENS/MODE key is set to SWITCH. However, with AQ6317 series instruments, if the measurement sensitivity is set to HIGH1, HIGH2, or HIGH3, a chopper that removes monochrometer stray light is activated. The instrument includes the following AQ6317-compatible mode commands that allow you to edit the settings of the chopper operation.

AQ6317 command to use to set the chopper operation

Control command

`CHOP*`

*: 0 = Chopper OFF, 2 = SWITCH mode

Query command

`CHOP?`

A return value: Same as the above

Appendix 6 Compatibility of Remote Commands of Legacy Models (AQ6370D, etc.)

List of Remote Commands Deleted or Changed from Legacy Models (AQ6370D, etc.)

Some remote commands used in legacy models such as the AQ6370D will be changed in the AQ6374E as follows due to feature compatibility.

Remote Command	Changes
:DISPlay[:WINDow]:SPLIt:HOLD:LOWer	Command deleted (feature not installed)
:DISPlay[:WINDow]:SPLIt:HOLD:UPPer	Command deleted (feature not installed)
:DISPlay[:WINDow]:SPLIt:POSition	Command deleted (feature not installed)
:DISPlay[:WINDow]:TRACe:Y[:SCALe]:DNUMber	Parameter: 12 invalid
:HCOPy:DESTination	Command deleted (feature not installed)
:HCOPy[:IMMediate]	Command deleted (feature not installed)
:HCOPy[:IMMediate]:FEED	Command deleted (feature not installed)
:HCOPy[:IMMediate]:FUNCtion:CALCulate:LIST	Command deleted (feature not installed)
:HCOPy[:IMMediate]:FUNCtion:MARKer:LIST	Command deleted (feature not installed)
:INITiate:SMODe	Parameter: SEG|4 invalid
:MEMory:CLEar	Command deleted (feature not installed)
:MEMory:EMPty?	Command deleted (feature not installed)
:MEMory:LOAD	Command deleted (feature not installed)
:MEMory:STORe	Command deleted (feature not installed)
:MMEMory:LOAD:MEMory	Command deleted (feature not installed)
:MMEMory:LOAD:TEMPlate	Command deleted (feature not installed)
:MMEMory:STORe:DATa:TYPE	Command deleted (only CSV data file format available)
:MMEMory:STORe:MEMory	Command deleted (feature not installed)
:MMEMory:STORe:TEMPlate	Command deleted (feature not installed)
:PROGram:EXECute	Command deleted (changed to :APPLication:PROGram:EXECute)
:SYSTem:DISPlay:TRANSparent	Command deleted (feature not installed)
:SYSTem:GRID:CUSTom:CLEar:ALL	Command deleted (custom grid feature changed)
:SYSTem:GRID:CUSTom:DELete	Command deleted (custom grid feature changed)
:SYSTem:GRID:CUSTom:INSert	Command deleted (custom grid feature changed)
:TRACe:TEMPlate	Command deleted (feature not installed)
:TRACe:TEMPlate:DATA	Command deleted (feature not installed)
:TRACe:TEMPlate:DATA:ADELete	Command deleted (feature not installed)
:TRACe:TEMPlate:DATA:ETYPe	Command deleted (feature not installed)
:TRACe:TEMPlate:DATA:MODE	Command deleted (feature not installed)
:TRACe:TEMPlate:DISPlay	Command deleted (feature not installed)
:TRACe:TEMPlate:GONogo	Command deleted (feature not installed)
:TRACe:TEMPlate:LEVel:SHIFt	Command deleted (feature not installed)
:TRACe:TEMPlate:RESult?	Command deleted (feature not installed)
:TRACe:TEMPlate:TTYPe	Command deleted (feature not installed)
:TRACe:TEMPlate:WAVelength:SHIFt	Command deleted (feature not installed)

Appendix 7 Compatibility of Program Commands and Variables of Legacy Models (AQ6370D, etc.)

List of Program Commands Deleted or Changed from Legacy Models (AQ6370D, etc.)

When a program file (.PG*) created with a legacy model such as AQ6370D is loaded into this instrument, some program commands will be changed as follows due to feature compatibility.

Command	Changes
[Sweep]	
SEGMENT MEASURE	Command deleted (feature not installed)
SEGMENT POINT *****	Command deleted (feature not installed)
[Level]	
Y SCALE DIVISION ##DIV	Parameter: 12 invalid
[Zoom]	
OVERVIEW SIZE LARGE	Command deleted (feature not installed)
OVERVIEW SIZE SMALL	Command deleted (feature not installed)
[Trace]	
SPLIT DISPLAY	Command deleted (feature not installed)
TRACE # UPPER	Command deleted (feature not installed)
TRACE # LOWER	Command deleted (feature not installed)
UPPER HOLD ###	Command deleted (feature not installed)
LOWER HOLD ###	Command deleted (feature not installed)
[Marker]	
MARKER LIST PRINT	Command deleted (feature not installed)
ADV MARKER ##,****.****nm	Execution only. Parameter input not available.
ADV MARKER ##,***.****THz	Execution only. Parameter input not available.
ADV MARKER ##,@@@@@	Execution only. Parameter input not available.
ADV MARKER TRACE ##,###	Execution only. Parameter input not available.
ADV MARKER SELECT ##,########	Execution only. Parameter input not available.
ADV MARKER INTEGRAL ##,***.*GHz	Execution only. Parameter input not available.
ADV MARKER PEAK SEARCH ##	Execution only. Parameter input not available.
ADV MARKER BOTTOM SEARCH ##	Execution only. Parameter input not available.
ADV MARKER NEXT SEARCH ##	Execution only. Parameter input not available.
ADV MARKER SEARCH RIGHT ##	Execution only. Parameter input not available.
ADV MARKER SEARCH LEFT ##	Execution only. Parameter input not available.
ADV MARKER BANDWIDTH **.*nm	Execution only. Parameter input not available.
ADV MARKER ALL CLEAR	Execution only.
[Analysis]	
PMD ANALYSIS	Command deleted (feature not installed)
PMD THRESH **.**dB	Command deleted (feature not installed)
GRID REFERENCE WL ****.****nm	Command deleted (feature not installed)
ANALYSIS RESULT PRINT	Command deleted (feature not installed)
RESULT WRITE INT:' '	Command deleted (feature not installed)
RESULT WRITE EXT:' '	Command deleted (feature not installed)
RESULT WRITE INT	Command deleted (feature not installed)
RESULT WRITE EXT	Command deleted (feature not installed)
RESULT WRITE INT @@	Command deleted (feature not installed)
RESULT WRITE EXT @@	Command deleted (feature not installed)

Command	Changes
[File]	
WRITE MEMORY ** INT:' '	Command deleted (feature not installed)
WRITE MEMORY ** EXT:' '	Command deleted (feature not installed)
WRITE MEMORY ** INT	Command deleted (feature not installed)
WRITE MEMORY ** EXT	Command deleted (feature not installed)
WRITE MEMORY ** INT @@	Command deleted (feature not installed)
WRITE MEMORY ** EXT @@	Command deleted (feature not installed)
GRAPH TYPE:TIFF	Command deleted (feature not installed)
DATA WRITE:DT8	Command deleted (only CSV data file format available)
READ MEMORY ** INT:' '	Command deleted (feature not installed)
READ MEMORY ** EXT:' '	Command deleted (feature not installed)
READ MEMORY ** INT @@	Command deleted (feature not installed)
READ MEMORY ** EXT @@	Command deleted (feature not installed)
READ DATA INT:' '	Command deleted (feature not installed)
READ DATA EXT:' '	Command deleted (feature not installed)
READ DATA INT @@	Command deleted (feature not installed)
READ DATA EXT @@	Command deleted (feature not installed)
READ TEMPLATE EXT:' '	Command deleted (feature not installed)
READ TARGET LINE EXT:' '	Command deleted (feature not installed)
[Memmory]	
SAVE #->MEMORY **	Command deleted (feature not installed)
SAVE #->MEMORY @	Command deleted (feature not installed)
RECALL MEMORY **->#	Command deleted (feature not installed)
RECALL MEMORY @->#	Command deleted (feature not installed)
CLEAR MEMORY **	Command deleted (feature not installed)
CLEAR MEMORY @	Command deleted (feature not installed)
[System]	
REMOTE INTERFACE:RS-232	Command deleted (feature not installed)
SELECT COLOR *	Only parameter 1 is valid.
WINDOW TRANSPARENT ###	Command deleted (feature not installed)
KEY MAP SELECT ###	Command deleted (feature not installed)
CAL OUTPUT ###	Command deleted (feature not installed)
[Advance]	
TEMPLATE GO/NO GO ###	Command deleted (feature not installed)
TEMPLATE DISPLAY ###	Command deleted (feature not installed)
TEMPLATE DISPLAY UPPER ###	Command deleted (feature not installed)
TEMPLATE DISPLAY LOWER ###	Command deleted (feature not installed)
TEMPLATE DISPLAY TARGET ###	Command deleted (feature not installed)
TEMPLATE TEST TYPE UPPER	Command deleted (feature not installed)
TEMPLATE TEST TYPE LOWER	Command deleted (feature not installed)
TEMPLATE TEST TYPE UP & LOW	Command deleted (feature not installed)
TEMPLATE WL SHIFT ****.***nm	Command deleted (feature not installed)
TEMPLATE LEVEL SHIFT ***.**dB	Command deleted (feature not installed)

Command	Changes
[Special Command]	
PRINTER FEED **	Command deleted (feature not installed)
PRINT '----------'	Command deleted (feature not installed)
PRINT @@@@@	Command deleted (feature not installed)
PRINT @@@@@;	Command deleted (feature not installed)
PRINT DATA AREA	Command deleted (feature not installed)
PRINT OUTPUT WINDOW	Command deleted (feature not installed)
SEND **'----------'	Command deleted (feature not installed)
SEND **'----------';@	Command deleted (feature not installed)
SEND **'-----';@;'-----'	Command deleted (feature not installed)
RECEIVE **;@$	Command deleted (feature not installed)
RESET OPTION	Command deleted (feature not installed)
SPOLL **;S	Command deleted (feature not installed)
SEND RS-232'----------'	Command deleted (feature not installed)
SEND RS-232'----------';@	Command deleted (feature not installed)
SEND RS-232'-----';@;'-----'	Command deleted (feature not installed)
SENDR RS-232'----------';@$	Command deleted (feature not installed)

List of Program Variables Deleted or Changed from Legacy Models (AQ6370D, etc.)

When a program file (.PG*) created with a legacy model such as AQ6370D is loaded into this instrument, the handling of some program variables will be changed as follows due to feature compatibility.

Variable Name	Changes
[ADVANCED MARKER variables]	
WAM1	Replaced by variable W1
WAM2	Replaced by variable W2
WAM3	Variable deleted (feature not installed)
WAM4	Variable deleted (feature not installed)
WAM2-WAM1	Replaced by variable W2-W1
WAM4-WAM3	Variable deleted (feature not installed)
LAM1	Replaced by variable L1
LAM2	Replaced by variable L2
LAM3	Variable deleted (feature not installed)
LAM4	Variable deleted (feature not installed)
LAM2-LAM1	Replaced by variable L2-L1
LAM4-LAM3	Variable deleted (feature not installed)

Index

S

T

U

V

Z