LCR METER

ZM2376

Instruction Manual (Remote Control)

NF Corporation

DA00042209-001

LCR METER

ZM2376

Instruction Manual (Remote Control)

Registered Trademarks

National Instruments and LabVIEW are registered trademarks of National Instruments Corporation in the United States.

Other company names and product names used in this Instruction Manual may be trademarks or registered trademarks of respective companies.

Preface

This instruction manual describes the remote control of ZM2376.

- The following instruction manuals are provided for ZM2376.
  - ZM2376 Instruction Manual (Basics)
    This manual describes basic items such as how to operate ZM2376 from the panel, specifications, and maintenance.
  - ZM2376 Instruction Manual (Remote Control)
    This manual describes the remote control of ZM2376.
    It includes the description of standard commands.
    Standard commands are used in the operation mode 0 (default value).
  - ZM2376 Instruction Manual (Alternative Commands)
    This manual describes the alternative commands of ZM2376.
    Alternative commands are used in the operation mode 1.
    If you find the alternative commands easier to use than the standard commands, use them.
    However, the alternative commands provide limited functions.

"ZM2376 Instruction Manual (Remote Control)" and "ZM2376 Instruction Manual (Alternative Commands)" are included in the attached CD-ROM.

It also includes sample programs that control the ZM2376. Sample programs combine typical programming languages and GPIB, USB, RS-232, and LAN (optional) interfaces. For details, see the documentation of attached CD-ROM.

■ This instruction manual has the following chapters.

1. PREPARATIONS BEFORE USE

   Describes the setting and precautions of the interfaces.

2. SWITCHING BETWEEN REMOTE STATE AND LOCAL STATE

   Describes the switching between remote operation and local operation.

3. RESPONSE TO INTERFACE MESSAGE

   Lists the responses to main IEEE-488.1 interface messages.

4. COMMAND LIST AND COMMAND TREE

   Describes the overview of all the commands.

5. COMMAND EXPLANATION

   Describes the details of individual commands.

6. STATUS SYSTEM

   Describes the status system.

7. TRIGGER SYSTEM

   Describes the overview of the trigger system and the measurement procedure of the remote control.

8. IMPORTING DATA USING MEASURED DATA BUFFER

   Describes the measurement procedure using the measured data buffer.

9. ERROR MESSAGES

   Describes the error messages of the remote control.

Table of Contents

Page

Attached Figures and Tables

Page

1. PREPARATIONS BEFORE USE

ZM2376 can be controlled remotely by USB, RS-232, GPIB, or LAN (optional).
By sending program messages from the controller it is possible to control the instrument similarly to panel operations and to receive measured values or setting status as response messages.
The connectors of respective interfaces are located on the rear panel of the ZM2376.

1.1 Remote Control Interface Selection

For ZM2376, use one of USB, RS-232, GPIB, and LAN as the remote control interface. Plural interfaces cannot be used at the same time.
The setting of remote control interface is made from the system setting menu.

SHIFT + [SYSTEM]

Press the SHIFT + [SYSTEM] keys to display the system setting menu.

SYSTEM settings	
0)INTERFACE 1)BEEPER >NEXT	Options (first page)

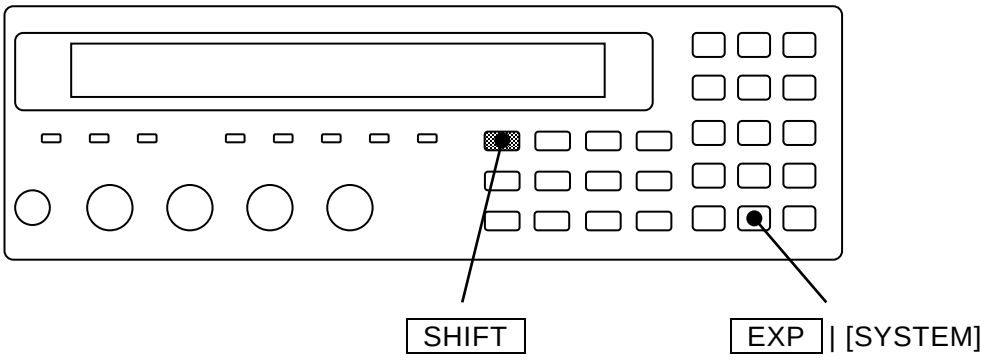
Select INTERFACE with the 0 key to display the remote control interface setting menu.

INTERFACE:USB	Current setting
0)USB 1)RS-232 2)GPIB 3)LAN	Options

USB	Selects USB (initial value) and displays the USB confirmation screen.
RS-232	Selects RS-232 and displays the RS-232 setting menu.
GPIB	Selects GPIB and displays the GPIB address setting menu.
LAN	Selects LAN (optional) and displays the LAN setting menu. When the option is not furnished, the option is not displayed.

Select either one with a numeric key.

Perform the EXIT operation to return to one-previous menu.



1.2 Outline of USB

1.2.1 Preparation of Controller

Prepare a controller (control computer) equipped with the USB interface when using the USB interface.

Install the USBTMC driver in the controller. Normally, this driver supports the Subclass USB488, so that almost same control as GPIB can be executed through the USB.

USBTMC: Universal Serial Bus Test and Measurement Class

This driver is included in the hardware products or software products of respective companies that supply the VISA library. The user who does not have the license of VISA library must get it additionally.

VISA: Virtual Instrument Software Architecture

Some of the companies that supply the VISA library (in random order, as of creation of this document)

National Instruments Corporation

Agilent Technologies, Inc.

Tektronix, Inc.

The VISA library enables unified operations whichever interface, USB, RS-232, GPIB, or LAN , is used as long as the library supports them.

1.2.2 Preparation of ZM2376

If USB is selected with the remote control interface setting menu, the USB confirmation screen is displayed as shown below.

USB	**Vendor:3402**	**Product:63**	**SN:9093251**
	0x0D4A	**0x003F**	**EXIT**

Vendor	Vendor ID=3402 (decimal notation): A number which indicates NF Corporation. It is 0x0D4A in hexadecimal notation.
Product	Product ID=63 (decimal notation): A product number which indicates ZM2376. It is 0x003F in hexadecimal notation.
SN	Serial Number=9093251 (example): 7-digit serial number uniquely assigned to each instrument.

Perform the EXIT operation to return to one-previous menu.

1.2.3 USB Device Identification

Connect the ZM2376 to the USB connector of the computer using commercially available USB cable. Connection via USB hub may result in an operation failure.

ZM2376 is automatically identified by connecting it with USB to the computer on which USBTMC class driver is installed.

The ZM2376 in the system is identified with the Vendor ID, Product ID, and Serial Number displayed on the USB confirmation screen. Use these values when specifying the instrument manually due to, for instance, automatic recognition error.

1.3 Outline of RS-232

1.3.1 Preparation of Controller

Prepare a controller (control computer) equipped with the serial communication (RS-232) connector when using the RS-232 interface.

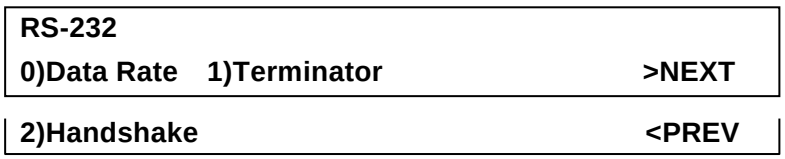
Adjust the following parameters between ZM2376 and controller:

- Data rate 4800 to 230400 bps
- Data length 8 bits
- Stop bit length 1 at transmission, 1 at reception
- Parity None
- Handshake None / Software / Hardware
- Terminator CR / LF / CR LF

Since the data length, stop bit length, and parity are fixed in ZM2376, adjust the settings on the controller side.

1.3.2 Preparation of ZM2376

If RS-232 is selected with the remote control interface setting menu, the RS-232 setting menu is displayed as shown below. Go to the sub menu for setting.



Options (first page)

Options (second page)

Data Rate	Displays the data rate setting menu.
Terminator	Displays the message terminator setting menu when data is transmitted from ZM2376.
Handshake	Displays the handshake setting menu.

Perform the EXIT operation to return to one-previous menu.

■ Data rate

The RS-232 data rate setting menu is as shown follows.

RS-232 Data rate: 9600bps	Current setting
0)4800 1)9600 2)19200 3)38400 >NEXT	Options (first page)
4)57600 5)115200 6)230400 <PREV	Options (second page)

Perform the EXIT operation to return to one-previous menu.

After setting, the results are displayed for a short time and return to one-previous menu.

The data rate is common to the transmission and reception.
Select the rate at which the communication can be made stably within the data rate range provided by both ZM2376 and controller. Even if a selection is made, when the data rate exceeds 19200 bps, stable communication may not be performed due to the restrictions of cable. For high speed communication, a short cable having low capacitance should be used.

■ Message terminator

A terminator indicating the termination is required at the end of a set of commands or response.
The RS-232 message terminator setting menu is as shown below.

RS-232 Terminator:LF	Current setting
0)CR 1)LF 2)CR LF	Options

CR	Uses one character CR (Carriage Return) as a terminator.
LF	Uses one character LF (Line Feed) as a terminator.
CR LF	Uses two characters CR and LF as a terminator.

Perform the EXIT operation to return to one-previous menu.

• When ZM2376 transmits data

The set terminator is added at the end of response message.

• When ZM2376 receives data

The command is executed when CR or LF, whichever comes first, is received.

After setting, the results are displayed for a short time and return to one-previous menu.

■ Handshake (flow control)

RS-232 Handshake setting menu is as shown follows.

RS-232 Handshake:SOFT	Current setting
0)OFF 1)SOFT 2)HARD	Options

OFF	No handshake (initial value)
SOFT	Software handshake (XON, XOFF) The communication is controlled with the control codes (XON, XOFF). Sure communication can be performed even with the cable connected to TxD, RxD, and GND only. However, binary data cannot be transferred. Also, effective speed may be lowered.
HARD	Hardware handshake (RTS, CTS) The communication is controlled with hardware control wires (RTS, CTS).

Perform the EXIT operation to return to one-previous menu.

After setting, the results are displayed for a short time and return to one-previous menu.
If the handshake is enabled, the transmission is suspended when the reception buffer becomes almost full, and it restarts when a space is made in the reception buffer.

1.3.3 Connection

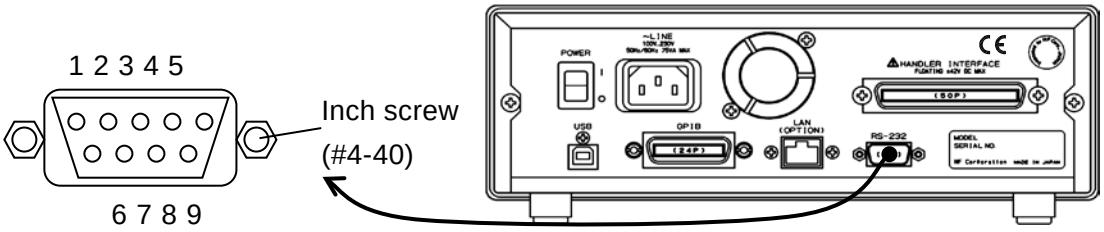
Prepare commercially available connection cable separately. When connecting to the serial interface of personal computer, the following cable can be used.

Cable specification: D-Sub, 9 pins, female – female, interlink connection, inch screw.
To avoid malfunction by electromagnetic noise radiation or noise, the shielded cable must be used.

The communication can be made, provided that at least three cables of RxD, TxD, and GND are connected.

For the hardware handshake, RTS and CTS are required.

To utilize the hardware handshake, use the interlink cable (Figure 1-1 (b)). For cross or reverse connection, a type in which adjacent pins 7 and 8 are connected (Figure 1-1 (c)) is possible. With this type, the communication can be made, but it cannot be used for the hardware handshake.



(a) RS-232 connector on the rear panel

ZM2376	
Signal	Pin No.
-	1
RxD	2
TxD	3
DTR	4
GND	5
-	6
RTS	7
CTS	8
-	9
	Frame

Personal computer (PC/AT, etc.)	
Pin No.	Signal
1	-
2	RxD
3	TxD
4	DTR
5	GND
6	-
7	RTS
8	CTS
9	-
Frame	

(b)Interlink connection

1	1
2	2
3	3
4	4
5	5
6	6
7	7
8	8
9	9

(c)Other cross connection

Figure 1-1 RS-232 cable connection diagram

1.3.4 Restrictions and Notes

- RS-232 is connected to the controller in one-to-one relation.
  Multiple instruments cannot be connected in parallel to one port.

- GPIB own functions such as SRQ and device clear cannot be used.
  If the instrument does not respond to commands due to communication path error or command misuse, turn off the power once, and turn it on again. Alternatively, switch to another interface once, and select RS-232 again.

- Clear the reception buffer before starting the communication.
  With the RS-232 communication path opened by the controller, turning on/off the power of ZM2376, or connecting/disconnecting the RS-232 connector causes abnormal data to enter the reception buffer in the controller. Accordingly, when the communication is started or restarted with the program on the controller, be sure to clear the reception buffer of the controller (for example, to initialize the communication) before performing usual operation.

1.4 Outline of GPIB

The GPIB interface was designed to be used in favorable environment. Avoid the use in a place with much noise.

1.4.1 Preparation of Controller

Install a commercially available GPIB interface card, etc. on the controller (control computer) and connect the controller with ZM2376 using the GPIB cable. For the GPIB driver software, refer to the instruction manual of the GPIB interface card you use.

1.4.2 Preparation of ZM2376

For the GPIB, the instrument in the system is identified with the instrument unique address. Set a unique GPIB address to each instrument. The GPIB address is set through the procedure given below.

■ GPIB address setting

Select GPIB with the remote control interface setting menu, and the GPIB address setting menu as shown below is displayed.

GPIB Address: 2	Current setting
0 to 30	Available value ranges

Perform the EXIT operation to return to one-previous menu.

After setting, the results are displayed for a short time and return to one-previous menu.

■ Message terminator

A terminator indicating the termination is required at the end of a set of commands or response.
The response message terminator that ZM2376 transmits is fixed to LF^EOI.
The program message terminator that ZM2376 receives is either of the followings:

- LF Line Feed code
- LF^EOI LF accompanied by EOI (END message)
- (Last code)^EOI EOI (END message) added to the last code

1.4.3 Precautions on Use of GPIB

- Connect or disconnect the GPIB connector with all instruments connected to the bus turned off.
- Turn on the power of all instruments connected to the bus when using the GPIB.
- The number of instruments connectable to one bus via GPIB is maximum 15 units including the controller.
  Also, the cable length is limited as follows:
  - Total cable length ≤ ((2 m × number of devices) or 20 m, whichever short)
  - Length of one cable ≤ 4 m
- For GPIB address, set a different value for each instrument. If plural instruments having same address exist on one bus, the instruments may be damaged due to the conflict of output.

1.4.4 Basic Specifications of GPIB

- GPIB conforming standards

  IEEE Std 488.1-1987, IEEE Std 488.2-1992

- IEEE Std 488.1-1987 interface functions

  SH1 Source Handshake full functions provided
  AH1 Acceptor Handshake full functions provided
  T6 Basic Talker, Serial Poll, and talker cancel function by listener-addressed provided
  Talk-Only function not provided
  L4 Basic Listener function and listener cancel function by talker-addressed provided
  Listen-Only function not provided
  SR1 Service Request full functions provided
  RL1 Remote Local full functions provided
  PP0 Parallel Poll function not provided
  DC1 Device Clear full functions provided
  DT1 Device Trigger full functions provided
  C0 Controller function not provided
  E1 Open collector drive

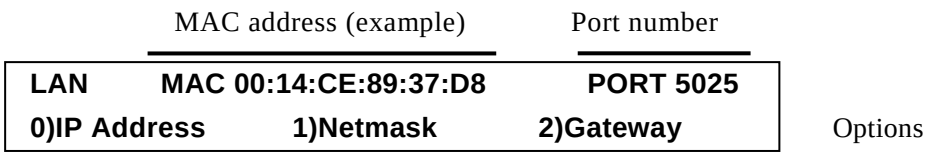
1.5 Outline of LAN

1.5.1 Preparation of Controller

Prepare a controller (control computer) equipped with the LAN interface when using the LAN interface (optional). ZM2376 can communicate using the TCP/IP protocol.

1.5.2 Preparation of ZM2376

If LAN is selected with the remote control interface setting menu, the LAN setting menu is displayed as shown below.



IP Address	Displays the IP address setting menu.
Netmask	Displays the subnet mask setting menu.
Gateway	Displays the default gateway setting menu.

Perform the EXIT operation to return to one-previous menu.

■ MAC address

Device unique address (physical address), 8-bit separated hexadecimal notation. MAC address cannot be changed.

■ Port number

Port number is used when ZM2376 communicates in the TCP protocol. Port number is fixed to 5025 (decimal notation) and cannot be changed.

ZM2376 does not support the automatic IP address assignment by DHCP. For network connection, consult the administrator to set the IP address, subnet mask, and default gateway. A wrong setting may cause interference to network communications.

IP Address: IP address setting menu

IP Address: 192.168.0.1	Current value
0.0.0.0 to 255.255.255.255	Available value ranges

Set an address (logical address) to identify the instrument in IP (Internet Protocol). 192.168.0.0 to 192.168.255.255 are private IP addresses which can be used freely in a minor local network (class C).

Netmask: Subnet mask setting menu

Subnet Mask: 255.255.255.0	Current value
0.0.0.0 to 255.255.255.255	Available value ranges

Set a mask to separate the upper network address from the lower host address in the IP address.

Gateway: Default gateway setting menu

Default Gateway: 0.0.0.0	Current value
0.0.0.0 to 255.255.255.255	Available value ranges

Set the IP address of the gateway (repeater) used implicitly when accessing an external network.

A parameter (32-bit) can be entered by pressing the ENTR key every 8 bits (0 to 255 in decimal notation). Finally, press the ENTR key again to set the value.
After setting, the results are displayed for a short time and return to one-previous menu.
Alternatively, perform the EXIT operation to return to one-previous menu.

1.5.3 Connection

When connecting ZM2376 to the network (or hub), a straight cable is used. However, if the hub can automatically recognize a straight and cross cables, a cross cable can also be used.
When connecting to a peer PC, a cross cable is used.

1.5.4 Restrictions and Notes

- GPIB own functions such as SRQ and device clear cannot be used.
  If the instrument does not respond to commands due to communication path error or command misuse, turn off the power once, and turn it on again. Alternatively, switch to another interface once, and select LAN again.

1.6 Precautions on Communication

■ Input buffer

- The sent commands are stored in the input buffer once, and they are interpreted and executed in order.
  The input buffer capacity is 1 KBytes (K=1024). Even program messages exceeding this size are all interpreted and executed in order.
- If a command out of the specification is found during interpretation or execution, an error occurs and after that, the command is not executed until the program message terminator is reached.

■ Output buffer

- The output buffer capacity is 64 KBytes (K=1024).
- If exceeding the maximum capacity, the output buffer is cleared and the query error bit of standard event status register is set to “1”. After that, the command interpretation and execution are performed as usual, but the response messages generated until the program message terminator is reached are all dumped.

■ Error queue

- The maximum length of error queue is 16.
- If exceeding this length, the last error in the queue is replaced with error “Queue overflow”. Error messages after that are dumped. Error messages up to the 15th error are retained.

■ Program message terminator

When a command is sent from the controller, add LF (Line Feed, 0A hex) at the end of transmitted character string as a program message terminator. Also, add EOI (END message) to the last byte. Sending a command without adding LF or EOI causes a malfunction in some instruments.

The program message terminator may not be output unless the program message terminator is specified in addition to the command body, depending on the driver software used in the computer for control. Though a new line (NL) may be expressed in place of line feed (LF), the binary code is same.

The RS-232 and LAN do not have the concept of END message and therefore they do not need EOI to be added.

■ Restrictions on RS-232 and LAN

GPIB own functions cannot be used. The following shows some examples.

Receiving the device clear (DCL, SDC) message

Receiving the GTL (Go To Local) message

Receiving the LLO (Local Lockout) message

Receiving the GET (Group Execute Trigger) message

Receiving the REN (Remote Enable) message

Receiving the SRQ (Service Request) message

Serial poll (receive SPE/SPD and send the status byte)

Sending the END message (EOI signal as a message terminator)

2. SWITCHING BETWEEN REMOTE STATE AND LOCAL STATE

In relation to the remote control, ZM2376 has the remote state and the local state.
In the local state, all panel operations are enabled.
In the remote state, the panel operations are disabled, except the operation to return to the local.

■ Selecting the remote state

Normally, the operation from USB or GPIB sets the remote state. This is caused by the GPIB driver's function on the controller side. According to the communication standard, if the REN message is set to "true" and the instrument is specified as a listener, that instrument becomes remote state.

■ Selecting the local state

Press the LOCAL key on the front panel, and the remote state is switched to the local state (except for local lockout).
Or, send the GTL command from the controller or return the REN line to "false", so that the local state can be set. If the GPIB cable is disconnected, the REN line becomes "false" and thus the local state is returned. Similarly, disconnecting the USB cable causes the instrument to be returned to the local state.

■ Disabling local operations from the panel

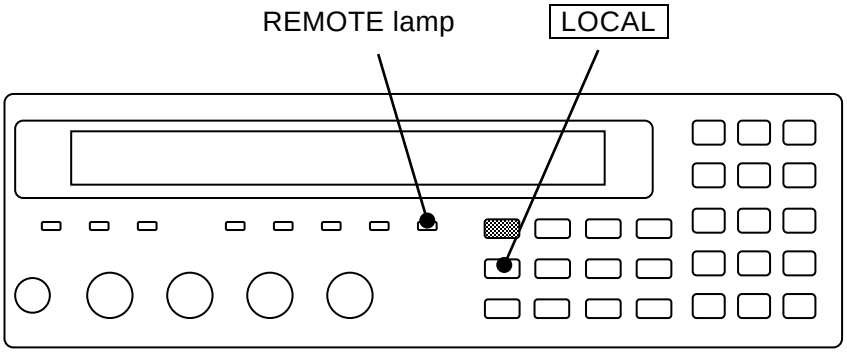
Accidental local operations can be disabled by specifying the local lockout from the controller. During the local lockout, the instrument cannot return to the local state even if the LOCAL key is pressed.
Even during the local lockout, the local state can be returned from the controller.

■ Remote/local operation of RS-232 and LAN

Send a command to the ZM2376, and the ZM2376 becomes remote state.
Press the LOCAL key to return to the local state, and the operation from the panel is enabled.
The local lockout function cannot be used. If you want to disable the panel operation, use the key lock function together.

REMOTE lamp

In the remote state, the REMOTE lamp lights up.



3. RESPONSE TO INTERFACE MESSAGE

The responses to main IEEE-488.1 interface messages are listed as below.

Table 3-1 Responses to interface messages

Message	Function
IFC	**< InterFace Clear >** Initializes the GPIB interface. Releases the specified listener and talker.
DCL,SDC	**< Device CLear >**, **< Selected Device Clear >** Clears the input buffer and stops interpretation and execution of a command. Clears the output buffer and clears the bit 4 (MAV) of status byte register. Releases the overlap command queuing by *WAI, *OPC, and *OPC? commands. Releases the trigger waiting state by READ? query.
LLO	**< Local LockOut** > Disables the transition from remote state to local state with the LOCAL key.
GTL	**< Go To Local >** Selects the local state.
GET	**< Group Execute Trig**ger > Executes the trigger. Same function as *TRG command.

How to send an interface message from the controller is different depending on the device driver. For details, see the instruction manual of each driver.

These functions cannot be used with RS-232 and LAN.

4. COMMAND LIST AND COMMAND TREE

ZM2376 commands are separated into the common commands defined by IEEE488.2 and subsystem commands which reflect instrument-specific functions.
The ZM2376 has different command systems depending on the operation mode. This section describes the standard commands. For details on other commands, refer to "ZM2376 Instruction Manual (Alternative Commands)".
Common commands that ZM2376 provides are listed in Table 4-1. Also, subsystem commands of ZM2376 are listed in Table 4-2.
The meanings of symbols used in Table 4-1 and Table 4-2 are as follows:

- Keywords shown in square brackets ([ ]) are omissible.
- The vertical bar (|) indicates the possibility to select a keyword from several keywords.
- It is possible to omit the lowercase part of each keyword.

Table 4-1 Common command list

Command	Name	Function
*CLS	Clear Status Command	Clears status data.
*ESE *ESE?	Standard Event Status Enable Command / Query	Sets/queries the standard event status enable register.
*ESR?	Standard Event Status Register Query	Queries the standard event status register contents.
*IDN?	Identification Query	Queries the instrument identification information (model name etc).
*OPC *OPC?	Operation Complete Command / Query	Requests to set the standard event status register's OPC bit to 1 when all command operations are completed. In response to query, it returns 1 when all operations are completed.
*OPT?	Option Identification Query	Queries the built-in options.
*RCL	Recall Command	Recalls the contents of specified setting memory.
*RST	Reset Command	Resets the instrument to restore the initial setting values.
*SAV	Save Command	Saves current settings in the specified setting memory.
*SRE *SRE?	Service Request Enable Command / Query	Sets/queries the service request enable register.
*STB?	Read Status Byte Query	Queries the status byte.
*TRG	Trigger Command	Applies a trigger to execute measurement once when the trigger source is BUS and the instrument waits for a trigger, and reads the measured data when measurement finished.
*TST?	Self-Test Query	Queries the self-diagnosis result. "0" (no error) is returned, if normal.
*WAI	Wait-to-Continue Command	Waits for executing the following commands until all of the preceding command operations are completed.

Table 4-2 Subsystem command list

Command	Function / Operation target
ABORt subsystem	
:ABORt	Aborts the measurement
CALCulate subsystem	
:CALCulate:COMParator:AUXBin	Enables the use of auxiliary bins
:CALCulate:COMParator:BEEPer:CONDition	Beeper sounding condition
:CALCulate:COMParator:BEEPer[:STATe]	Enables the use of beeper
:CALCulate:COMParator:CLEar	Initializes comparator settings
:CALCulate:COMParator:MODE	Comparator primary parameter comparison format
:CALCulate:COMParator:PRIMary:BIN{1|2|..|14}	Bin (primary parameter) lower & upper limit values
:CALCulate:COMParator:PRIMary:BIN{1|2|..|14}:STATe	Enables bin sorting (primary parameter)
:CALCulate:COMParator:PRIMary:NOMinal	Reference value for primary parameter deviation
:CALCulate:COMParator:SECondary:LIMit	Secondary parameter lower & upper limit values
:CALCulate:COMParator:SECondary:STATe	Enables secondary parameter comparison
:CALCulate:COMParator[:STATe]	Enables the use of comparator
:CALCulate:COMParator:EXTension[:STATe]	Enables bin extended function
:CALCulate:FORMat:AUTO[:STATe]	Enables measurement parameter auto selection
CALCulate1 subsystem	
:CALCulate1:CKIT:AUTO[:STATe]	Enables equivalent circuit auto selection
:CALCulate1:FORMat	Selection of Primary Parameters
:CALCulate1:LIMit:CLEar	Clears primary parameter comparison result
:CALCulate1:LIMit:FAIL?	Queries primary parameter comparison result
:CALCulate1:LIMit:LOWer[:DATA]	Primary parameter BIN1 lower limit value
:CALCulate1:LIMit:LOWer:STATe	Enables primary parameter BIN1 lower limit comparison
:CALCulate1:LIMit:STATe	Enables primary parameter BIN1 sorting
:CALCulate1:LIMit:UPPer[:DATA]	Primary parameter BIN1 upper limit value
:CALCulate1:LIMit:UPPer:STATe	Enables primary parameter BIN1 upper limit comparison
:CALCulate1:MATH:EXPRession:NAME	Primary parameter deviation display format
:CALCulate1:MATH:STATe	Enables primary parameter deviation display / output
CALCulate2 subsystem	
:CALCulate2:FORMat	Selection of secondary parameter
:CALCulate2:LIMit:CLEar	Clears secondary parameter comparison result
:CALCulate2:LIMit:FAIL?	Queries secondary parameter comparison result
:CALCulate2:LIMit:LOWer[:DATA]	Secondary parameter lower limit value
:CALCulate2:LIMit:LOWer:STATe	Enables secondary parameter lower limit comparison
:CALCulate2:LIMit:STATe	Enables secondary parameter comparison
:CALCulate2:LIMit:UPPer[:DATA]	Secondary parameter upper limit value
:CALCulate2:LIMit:UPPer:STATe	Enables secondary parameter upper limit comparison
:CALCulate2:MATH:EXPRession:NAME	Secondary parameter deviation display format
:CALCulate2:MATH:STATe	Enables secondary parameter deviation display / output

Supplement: The command that makes inquiry is called “query” and it terminates with a question mark. In this table, the query is omitted in the functions capable of making both setting and query.

Table 4-2 Subsystem command list

Command	Function / Operation target
CALCulate3 subsystem	
:CALCulate3:MATH:STATe	Enables IV monitor display
CALCulate4 subsystem	
:CALCulate4:MATH:STATe	Enables IV monitor display
CALibration subsystem	
:CALibration:CABLe	Cable Length Correction
DATA subsystem	
:DATA[:DATA]	Reference value for primary & secondary parameter deviation display
:DATA[:DATA]?	Queries reference value for deviation display, contents of data buffer, and IV monitored value
:DATA:FEED	Specify recording data in measured data buffer
:DATA:FEED:CONTrol	Enables recording in measured data buffer
:DATA:POINts	Measured data buffer virtual size and initialization
DISPlay subsystem	
:DISPlay[:WINDow][:STATe]	Enables measured value display (enabled at all times)
:DISPlay[:WINDow]:TEXT1:DIGit	Number of measured value display digits (fixed)
:DISPlay[:WINDow]:TEXT1:PAGE	Measured value/comparison result selection (both at all times)
:DISPlay[:WINDow]:TEXT2:PAGE	Auxiliary display item (function limited)
:DISPlay[:WINDow]:TEXT3[:PAGE]	Auxiliary display item
FETCh subsystem	
:FETCh?	Query of latest measured data
FORMat subsystem	
:FORMat[:DATA]	Measured data transfer format
INITiate subsystem	
:INITiate:CONTinuous	Continuous starting of trigger system
:INITiate[:IMMediate]	Starting of trigger system
LIST subsystem	
:LIST:FREQuency	Frequency of multi-measurement
:LIST:MEMory	Parameters other than frequency
:LIST:MODE	Multi-measurement mode
:LIST[:STATe]	Enables multi-measurement
READ subsystem	
:READ?	After waiting for a trigger, Query of the next measurement data
SENSe subsystem	
[:SENSe]:AVERage:COUNt	Averaging count
[:SENSe]:AVERage[:STATe]	Enables the averaging
[:SENSe]:CORRection:CKIT:STANdard1:FORMat	Format of OPEN correction value
[:SENSe]:CORRection:CKIT:STANdard2:FORMat	Format of SHORT correction value
[:SENSe]:CORRection:CKIT:STANdard3:FORMat	Format of LOAD correction value / standard value
[:SENSe]:CORRection:CKIT:STANdard3[:SPOT]	Standard value for spot LOAD correction
[:SENSe]:CORRection:COLLect[:ACQuire]	Measurement of correction value
[:SENSe]:CORRection:COLLect:METHod	Correction operating target
[:SENSe]:CORRection:DATA[:SPOT]	Correction value

Table 4-2 Subsystem command list

3/4

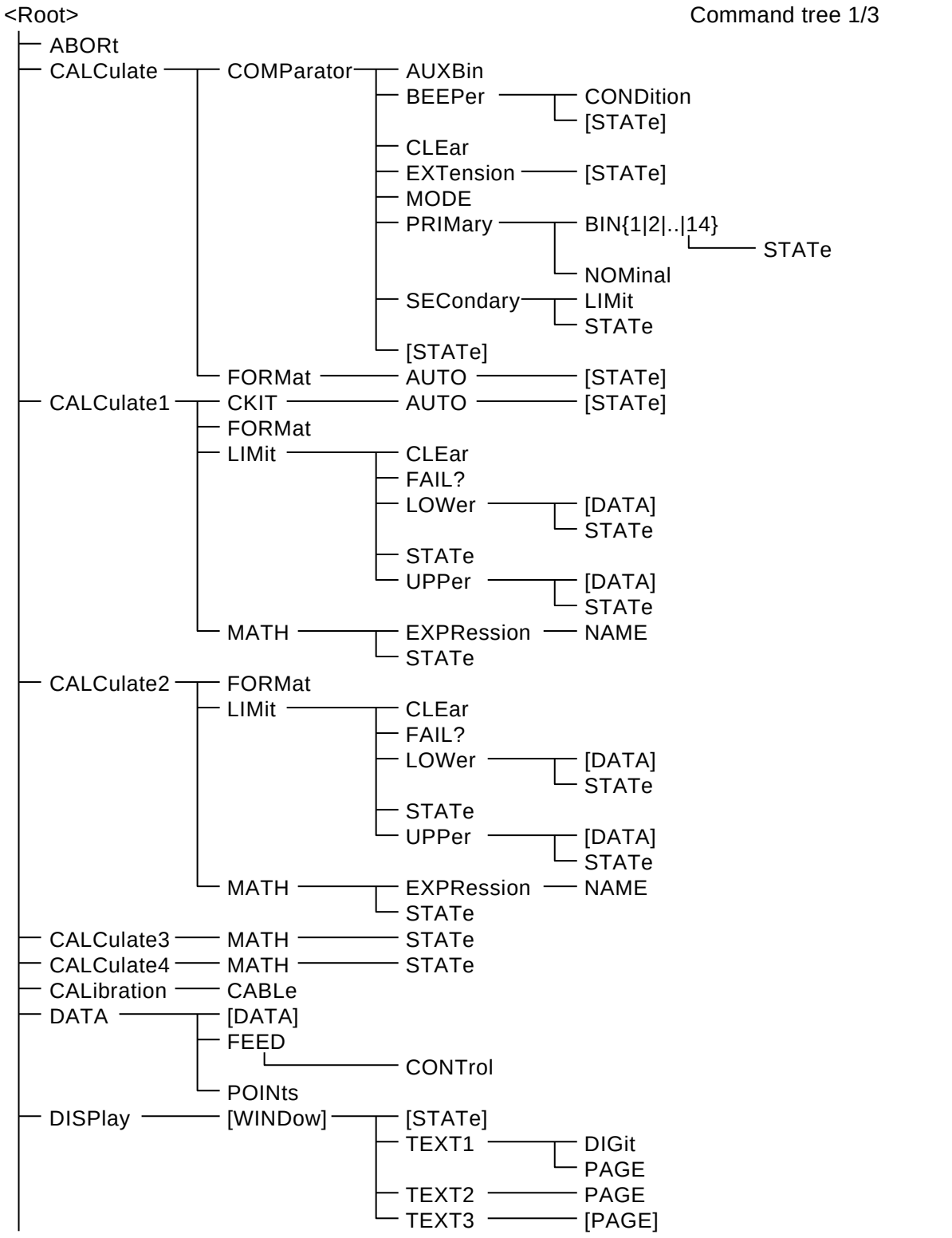
Command	Function / Operation target
SENSe subsystem (Continued)	
[:SENSe]:CORRection:LIMit:LOWer	Correction lower limit frequency
[:SENSe]:CORRection:LIMit:UPPer	Correction upper limit frequency
[:SENSe]:CORRection:LOAD[:STATe]	Enables LOAD correction
[:SENSe]:CORRection:OPEN[:STATe]	Enables OPEN correction
[:SENSe]:CORRection:SHORT[:STATe]	Enables SHORT correction
[:SENSe]:CORRection:SPOT[:STATe]	Enables SPOT correction measurement
[:SENSe]:CORRection[:STATe]	Enables correction in a lump
[:SENSe][:FIMPedance]:APERture[:MODE]	Measurement speed / measurement time
[:SENSe][:FIMPedance]:CONTact:VERify	Enables contact check
[:SENSe][:FIMPedance]:CREJect:LIMIt	Low capacitance limit
[:SENSe][:FIMPedance]:CREJect[:STATe]	Enables low capacitance check
[:SENSe][:FIMPedance]:RANGe:AUTO	Enables measurement range auto switching
[:SENSe][:FIMPedance]:RANGe[:UPPer]	Measurement range
[:SENSe]:FRESistance:RANGe:AUTO	Enables Rdc measurement range auto switching
[:SENSe]:FRESistance:RANGe[:UPPer]	Rdc measurement range
[:SENSe]:FUNCtion:CONCurrent	Enables execution of simultaneous measurements
[:SENSe]:FUNCtion[:ON]	Specifies measurement function
SOURce subsystem	
:SOURce:CURRent:ALC[:STATe]	Enables constant current drive
:SOURce:CURRent[:LEVel][:IMMediate][:AMPLitude]	Constant current level
:SOURce:FREQuency[:CW]	Measurement frequency
:SOURce:RESistance[:LIMit]:LOWer	Minimum output impedance
:SOURce:RESistance[:LIMit]:ZRANge	Selects Rd 25Ω -1/-100
:SOURce:VOLTage:ALC[:STATe]	Enables constant voltage drive
:SOURce:VOLTage[:LEVel][:IMMediate][:AMPLitude]	Measurement voltage level
:SOURce:VOLTage[:LEVel][:IMMediate]:OFFSet	DC bias voltage
:SOURce:VOLTage[:LEVel][:IMMediate]:OFFSet:AUTO	Selects auto DC balance function
:SOURce:VOLTage[:LEVel][:IMMediate]:OFFSet:STATe	Enables DC bias
:SOURce:VOLTage:MODE	Enables triggered drive
STATus subsystem	
:STATus:OPERation:CONDition?	Queries the condition (OPCR)
:STATus:OPERation:ENABle	Enables an event (OPEE)
:STATus:OPERation[:EVENt]?	Queries an event (OPER)
SYSTem subsystem	
:SYSTem:ADELay	Settling wait time during automatic adjustment
:SYSTem:ERRor?	Queries the content of error
:SYSTem:KLOCk	Enables key lock
:SYSTem:MEMory	Selects settings / correction value
:SYSTem:RCL	Recalls settings / correction value
:SYSTem:RST	Performs initialization including memory
:SYSTem:SAVE	Saves settings / correction value

Table 4-2 Subsystem command list

4/4

Command	Function / Operation target
TEST subsystem	
:TEST:HANDler:BIN	/BINxx test output
:TEST:HANDler:COMP	Comparison test output
:TEST:HANDler:KEYLock?	/KEY_LOCK input status
:TEST:HANDler:MODE	Enables test mode
:TEST:HANDler:RCL?	/RCLx input status
:TEST:HANDler:RCLValid?	/RCL-VALID input status
:TEST:HANDler:STATus:EOM	/EOM test output
:TEST:HANDler:STATus:INDex	/INDEX test output
:TEST:HANDler:STATus:NC	/NC test output
:TEST:HANDler:STATus:ERR	/ERR test output
:TEST:HANDler:TRIGger?	TRIG (/TRIG) input status
TRIGger subsystem	
:TRIGger:SOURce	Trigger source
:TRIGger[:IMMediate]	Trigger (measurement start)
:TRIGger:DELay	Trigger delay time

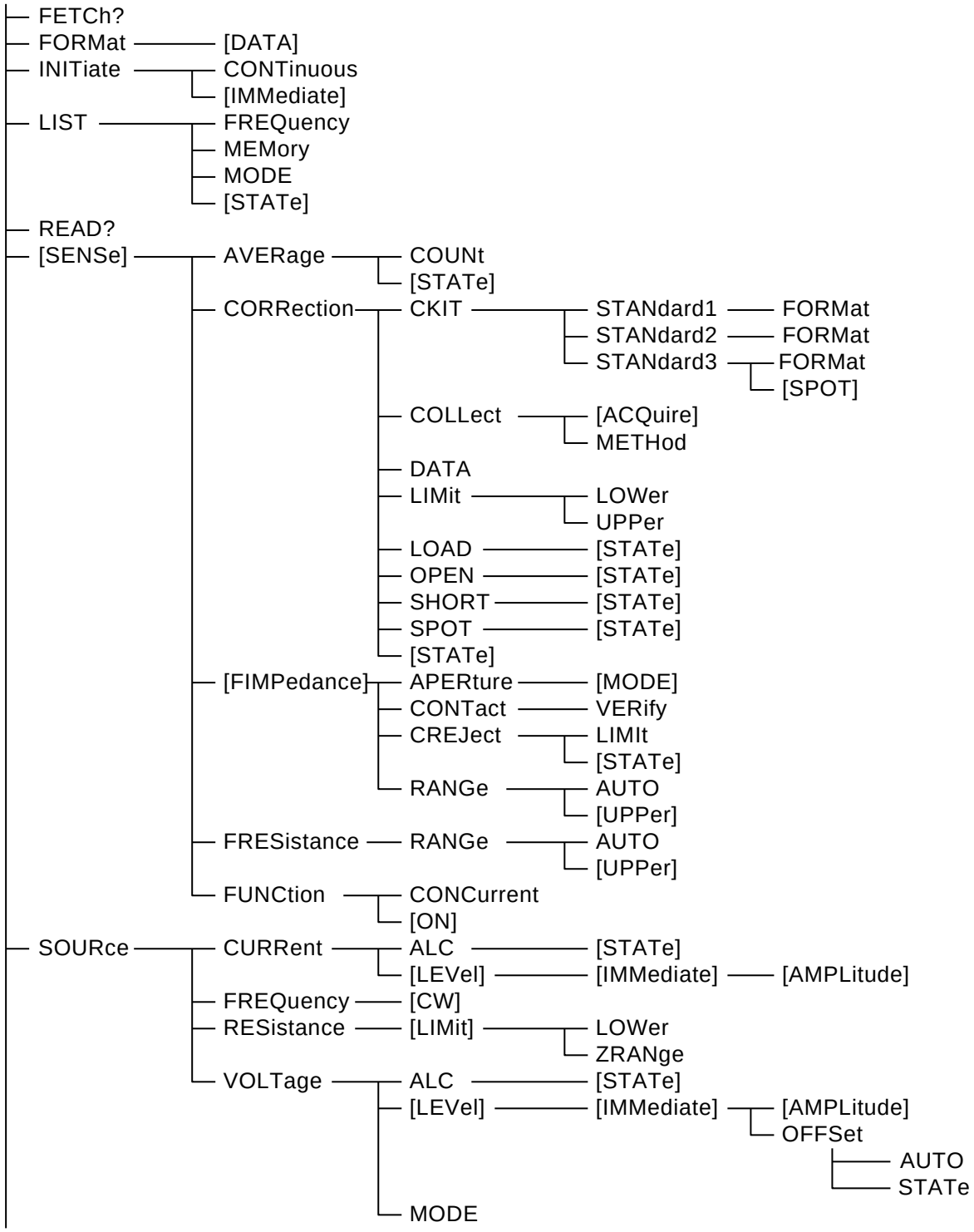
The subsystem command tree of ZM2376 is shown below.



Continued

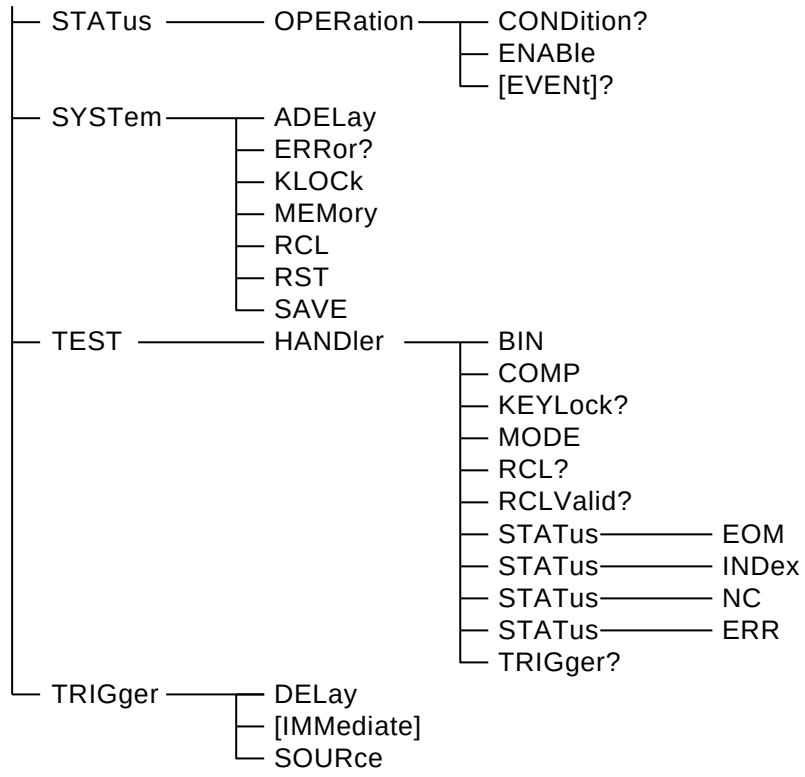
Square brackets (**[ ]**) indicate optional keywords.

Command tree 2/3



Continued

Command tree 3/3



5. COMMAND EXPLANATION

5.1 Outline of Command Language

Summary of terms used are explained below.

5.1.1 Subsystem Commands

Commands are divided into groups depending on their function. Subsystem commands are arranged in a hierarchy. The colon (:) is defined as the path separator.

5.1.2 Path Separator

The path separator (:) separates the current keyword from the keywords one level lower. In a command character string, each colon (:) indicates a displacement to the path one level below the current path.

When the colon (:) is found at the beginning of a command character string it means “to set the current path to root”. The current path is also set to root by the power on, *RST command, or message terminator. The beginning of a program message is always set to root. A colon (:) at the beginning of a command character string is omissible arbitrarily.

:SYST:KLOC 1

↑ ↑ ↑ ↑

① ② ③ ④

①Set current path to root (Omissible)

②SYSTem subsystem command (SYSTem is a root command)

③SYSTem subsystem attached KLOCk command

④A space is required between the header and the parameters

By separating the commands with a semicolon (;), plural command character strings can be put together into one program message.

:CALC1:MATH:STAT ON **;** **:CALC1:MATH:**EXPR:NAME PCNT

Command 1 Command 2

The above example shows one program message equivalent to the following two program messages.

:CALC1:MATH:STAT ON Current path after execution is **:CALC1:MATH:**

:CALC1:MATH:EXPR:NAME PCNT Current path after execution is **:CALC1:MATH:EXPR:**

If a colon (**:**) is omitted in the second and subsequent command character strings, the subsystem command on the same level is accessible without changing the current path.

:CALC1:MATH:STAT ON **;** EXPR:NAME PCNT In second command **:CALC1:MATH:** is omissible

In both cases, the program message terminator at the end is omitted.

5.1.3 Keywords Simplification

In this Instruction Manual, a keyword indicating a command or parameter is expressed with a combination of uppercase and lowercase alphabets. Uppercase characters are used for short form (abbreviation). Even if lowercase character part is all omitted, it has the same function as long form that includes all the lowercase character part. However, only a part of lowercase character part cannot be omitted.
Uppercase and lowercase characters are used only as an expedient means to explicit the form that can be shortened but instrument makes no distinction whatsoever between the two. It is possible to use both simultaneously.

Ex.) Command notation :CALCulate1:FORMat? (○ - Acceptable, × - Error)

→	:calculate1:format?	○ Long form, all lowercase character
	:Calc1:Form?	○ Short form, mixed uppercase/lowercase characters
	:**CALCUL**1:FORM?	× Wrong intermediary abbreviated form
	:CALC1:**FOR**?	× Omission over-abbreviated

5.1.4 Implicit Keywords

Keywords shown in square brackets ([ ]) indicate the implicit keywords and are omissible. The instrument performs same operation if all or part of implicit keyword is omitted or even if not omitted at all. In the following example, both command notations have same function to the instrument. The following examples show an instrument function using both ways.

Ex.) Command notation [:SENSe]:AVERage[:STATe] ON

→	:SENS:AVER:STAT ON	Implicit keyword is not omitted
	:AVER ON	Implicit keyword is all omitted

5.2 Overlap Commands and Sequential Commands

Some commands allow the following commands to be executed simultaneously during execution of them, and some commands do not allow.

■ Overlap commands

During execution of that command, the following command can be executed.

■ Sequential commands

After execution of that command finished, the following command is executed.

Note that executing multiple overlap commands will result in an operation failure.
Use *WAI, *OPC, and *OPC? commands when you do not want to execute the following commands or query until the execution of overlap commands finished.
The commands given below are overlap commands. Other commands are all sequential commands.
Overlap commands : [:SENSe]:CORRection:COLLect[:ACQuire]

5.3 Command Detailed Explanations

The function and syntax of all the commands shown in “Table 4-1 Common command list” and “Table 4-2 Subsystem command list” are explained below.

[Meaning of symbols]

- Keywords shown in square brackets (**[ ]**) are omissible.(Implicit Keywords)
- Curly braces (**{ }**) indicate parameters in the command character string.
- The vertical bar (|) indicates possibilities to select a keyword from several options.
- Comparison marks (< >) indicate that a parameter such as numerical values and character strings need to be set.

These symbols are used for description only. They are not used in actual commands.

Description [:SENSe][:FIMPedance]:RANGe:AUTO {ON|OFF|1|0}

Actual command :SENSe:FIMPedance:RANGe:AUTO ON

[Parameters data format]

Symbol	Format	ex.
NR1	Integer (numeric value)	123
NR2	Decimal point format no having exponent (numeric value)	0.075
NR3	Decimal point format having exponent (numeric value)	4.99E+06
CRD	Character strings	ALL
SRD	Character strings in double quotation marks	"No error"
bool	Logical value	ON, OFF, 1, 0

- Unless otherwise specified individually, a numeric value in any format is accepted and it is rounded to the nearest value if exceeding the specified resolution.
- Unless otherwise specified individually, a numeric value is set to minimum value if below the minimum value, or set to maximum value if above the maximum value.
- For numeric value parameters in which the description is made so that MAX / MIN can be used, a numeric value is set to maximum value if MAX (or MAXIMUM) is given, or set to minimum value if MIN (or MINIMUM) is given.
- When the response is a numeric value, and the range, resolution and unit are omitted, it is same as a setting. Unless otherwise specified individually, the mantissa of response in NR3 format is 6 digits.
- For the commands in which the parameter type varies depending on the format setting, even if a numeric value exceeding the range mentioned below is given, it is rounded to the same range as displayed on panel or resolution. For the range of actually set value, see the description of operation panel.
- Character strings in quotation marks are accepted in single or double quotation marks.

[Remarks]

- Both commands and queries are called commands here.
  The keyword attached with “**?**” at the end is a query.
- Response messages to commands do not come with headers.

5.3.1 Common Commands

***CLS**

Description: Clears the following statuses.
- Standard event status register
- Operation event register
- Status Byte
- Error queue

Also, the error display on the panel is reset.

Setting example: *CLS

Remarks: The *CLS command does not directly clears the status byte register. Except for the MAV bit and the RQS bit, the status byte is indirectly cleared. It is possible to indirectly clear the MAV bit by clearing the input buffer with a device clear. It is possible to clear the RQS bit by reading out the status with the serial pole.

***ESE <mask>**

***ESE?**

Description: Sets/Queries the standard event status enable register.

Parameter: <mask> {numeric value, range 0 to 255} Out of range value causes an error.

Details • • • ☞ "6.3 Standard Event Status"

Setting example: *ESE 255

Sets 255 to the standard event status enable register.

Response: <mask> {NR1, range 0 to 255}

Query example: *ESE?

Response example: +255

The content of standard event status enable register is 255.

***ESR?**

Description: Queries the standard event status register contents.

The query clears all the standard event status register bits to 0.

Response: <register contents> {NR1, range 0 to 255}

Details • • • ☞ "6.3 Standard Event Status"

Query example: *ESR?

Response example: +128

The content of standard event status register is 128.

***IDN?**

Description: Queries the model name etc.

Response: {"<name of manufacturer>, <model name>, <serial number>, <firmware version>"}
format SRD

Query example: *IDN?

Response example: "NF Corporation,ZM2376,9055552,Ver1.00"

***OPC**

***OPC?**

Description: Confirms that the execution of all the preceding commands is completed.

Setting example: *OPC
Makes setting so that OPC bit of the standard event status register is set to "1" when the execution of all the preceding commands is completed. By monitoring the status, the completion of command execution can be known.

Response: 1
1 is returned when the execution of all the preceding commands is completed.

Query example: *OPC?

Response example: 1
The execution of all commands is completed.

Remarks: The OPC bit of standard event status register is not cleared by *OPC?. To clear it, use the device clear, *CLS, or *RST command.
You can also generate SRQ when the OPC bit is changed to 1.

***OPT?**

Description: Queries options that are furnished in the instrument.

Response: {NR1, range {0|<option 1>[,<option 2>][,<option 3>]•••} }
Format of each field

0	No option is furnished.
1	LAN interface
Others	At present, no other options are provided.

Query example: *OPT?

Response example: +0
No option is furnished.

***RCL <memory number>**

Description: Recalls the settings from the specified setting memory.

Parameter: <memory number> {NR1, range 0 to 31}

Out of range value causes an error.

Setting example: *RCL 5

Recalls the settings from the setting memory number 5.

Remarks: To recall correction values, use the :SYSTem:RCL command.

***RST**

Description: Resets to initial setting state.

Details • • • ☞ ZM2376 Instruction Manual (Basics)
"Table 3-1 Setting items and initial values"

Setting example: *RST

Remarks: Beware of the following points:

• OPEN, SHORT, and LOAD correction values are initialized. When using previous correction values, save them in advance, and recall them after *RST.

• • • ☞ :SYSTem:SAVE, :SYSTem:RCL commands

• The initial value of trigger delay time is not zero.

***SAV <memory number>**

Description: Saves the currently used setting to the specified setting memory.

Parameter: <memory number> {NR1, range 0 to 31}

Out of range value causes an error.

Setting example: *SAV 5

Saves current settings in the setting memory number 5.

Remarks: To save correction values, use the :SYSTem:SAVE command.

***SRE <SRQ mask>**

***SRE?**

Description: Sets/Queries the service request enable register.

Parameter: <SRQ mask> {NR1, range 0 to 255}

Out of range value causes an error.

Details • • • ☞ "6.1 Outline of Status System"

Setting example: *SRE 128

Response: {NR1, range 0 to 255}

Query example: *SRE?

Response example: +128

The content of service request enable register is 128.

***STB?**

Description: Queries the content of status byte register.

Response: <register contents> {NR1, range 0 to 255}

Details • • • ☞ "6.2 Status Byte"

Query example: *STB?

Response example: +128

The content of status byte register is 128.

***TRG**

Description: When the instrument waits for a trigger, a trigger is applied to execute measurement once, and when the measurement finished, the measured data is read.

Setting example: *TRG

Applies a trigger, and reads the measured data when new measured data is obtained.

Response: <measurement status>, <primary parameter measured value>, <secondary parameter measured value>

[, {<comparator bin sorting result> | <limit comparison result> }]

Response message is same as :FETCh? query.

Details • • • ☞ :FETCh?

Remarks: When the trigger source is not BUS or when the instrument does not wait for a trigger, a trigger is not applied, causing an error (-211, "Trigger ignored"). When applying a trigger with the *TRG command or the interface message GET, the measured data is placed on the output buffer at the completion of measurement. Applying the next trigger without receiving this data causes an error (-410, "Query INTERRUPTED").

***TST?**

Description: Queries the self-diagnosis results.

ZM2376 conducts the self-diagnosis and returns its result.

Response: {NR1, range 0 to 4}

0: Pass (no abnormality is found)

1: Hardware failed (whole measurement circuits are faulty)

2: Oscillator failed (drive signal source is faulty)

3: Analyzer failed (voltage & current measurement section is faulty)

4: HF failed (error of high-frequency internal impedance bridge)

Query example: TST?

Response example: +0

No abnormality was found in the self-diagnosis.

***WAI**

Description:	Waits for executing the following commands until all of the preceding command operations are completed.
Example:	Overlap command 1 ; Overlap command 2 ; *WAI ; following command <program message terminator> After the execution of both overlap command 1 and overlap command 2 finished, the following commands are executed.
Remarks	The waiting caused by the *WAI command can be canceled with the Device Clear function.

5.3.2 Subsystem Commands

:ABORt

Description: Aborts the measurement and places the trigger system in idle state. When continuous initiation of trigger system is enabled, the instrument goes in trigger waiting state automatically. Further when the trigger source is set to INT (internal), a new trigger is applied automatically to start the measurement.
The measurement of OPEN, SHORT, and LOAD correction values is aborted.

Setting example: :ABOR

:CALCulate:COMParator:AUXBin {ON|OFF|1|0}

:CALCulate:COMParator:AUXBin?

Description: Sets/Queries whether S-NG (primary parameter is within the range but secondary parameter is out of range) is used as auxiliary bins (AUX_BIN) independent from OUT OF BINS when the bin sorting is performed by the comparator function.

Parameter: {ON|1} S-NG is used as auxiliary bins.
When judged as S-NG, OUT OF BINS is not output.

{OFF|0} S-NG is not used as auxiliary bins.
When judged as S-NG, both S-NG and OUT OF BINS are output. OUT OF BINS includes both primary parameter out of range and secondary parameter out of range.

Setting example: :CALC:COMP:AUXB ON
Sets so that S-NG is used as auxiliary bins.

Response: {1|0}

Query example: :CALC:COMP:AUXB?

Response example: 1
S-NG is used as auxiliary bins.

:CALCulate:COMParator:BEEPer:CONDition {FAIL|PASS}

:CALCulate:COMParator:BEEPer:CONDition?

Description: Sets/Queries whether the beeper sounds when the comparator result does not fall in BIN1..BIN14 or when the result falls in the range.
To sound the beeper, enable the beeper by giving additionally the :CALCulate:COMParator:BEEPer[:STATe] command.

Parameter: FAIL The beeper sounds when the comparison result is other than BIN1..BIN14 (namely, OUT OF BINS, S-NG, ERR).
In the limit comparison, the beeper sounds when the comparison result is not IN.
PASS The beeper sounds when the comparison result is within BIN1..BIN14.In the limit comparison, the beeper sounds when the comparison result is IN.

Setting example: :CALC:COMP:BEEP:COND FAIL
Sets so that the beeper sounds when the comparison result does not fall in BIN1 - BIN14.

Response: {FAIL|PASS}

Query example: :CALC:COMP:BEEP:COND?

Response example: FAIL
The beeper sounds when the comparison result does not fall in BIN1 - BIN14.

:CALCulate:COMParator:BEEPer[:STATe] {ON|OFF|1|0}

:CALCulate:COMParator:BEEPer[:STATe]?

Description: Sets or queries whether the beeper is enabled or disabled.
If disabled, the beeper does not sound regardless of comparator result.

Parameter: {ON|1} Enables the beeper.
{OFF|0} Disables the beeper.

Setting example: :CALC:COMP:BEEP ON
Enables the beeper.

Response: {1|0}

Query example: :CALC:COMP:BEEP?

Response example: 1
The beeper is enabled.

:CALCulate:COMParator:CLEar

Description: Initializes the comparator setting.

Details • • • ☞ ZM2376 Instruction Manual (Basics)
"Table 4-2 Initialization contents of comparator"

Setting example: :CALC:COMP:CLE
Initializes the comparator setting.

:CALCulate:COMParator:EXTension[:STATe] {ON|OFF|1|0}

:CALCulate:COMParator: EXTension[:STATe]?

Description: Sets/Queries whether the bin extended function of comparator functions is used or not.

Parameter: {ON|1} Enables the bin extended function (BIN10 - BIN14).
{OFF|0} Disables the bin extended function (BIN10 - BIN14).
When the bin extended function is enabled, BIN10 - BIN14 signals of handler interface can be used. Instead, the PHI and PLO signals are not output.
When the bin extended function is disabled, BIN10 - BIN14 sorting is not executed. Instead, the PHI and PLO signals of handler interface can be used.

Setting example: :CALC:COMP:EXT ON
Sets so that the bin extended function is enabled.

Response: {1|0}

Query example: :CALC:COMP:EXT?

Response example: 1
The bin extended function is enabled.

:CALCulate:COMParator:MODE {ABS|DEV|PCNT}

:CALCulate:COMParator:MODE?

Description: Sets/Queries the comparison format of primary parameters when the bin sorting is executed by the comparator function. This format is common to the display format of primary parameters.

Parameter: ABS Compares absolute value (original measured value).
DEV Compares deviation from reference value (= measured value – reference value).
PCNT Compares percent value of deviation (to reference value).

Setting example: :CALC:COMP:MODE PCNT
Sets so that the percent value of deviation is compared.

Response: {ABS|DEV|PCNT}

Query example: :CALC:COMP:MODE?

Response example: PCNT
The percent value of deviation is compared.

Remarks: The comparison format of secondary parameters follows the display format of primary parameters.

------ Absolute value ------

For ABS, the comparison is made to original measured value. A negative value is not converted into a positive value. Original measured value including a sign is used.

------ Unit of value ------

The unit of upper limit value, lower limit value, and reference value for deviation comparison of primary and secondary parameters is interpreted according to the parameter type or setting of deviation measurement. For example, if the capacitance C is displayed, the unit is interpreted as F (farad), or if a percent value of deviation is displayed, the unit is interpreted as %.

:CALCulate:COMParator:PRIMary:BIN{1|2|3|4|5|6|7|8|9|10|11|12|13|14} <lower limit value>, <upper limit value>

:CALCulate:COMParator:PRIMary:BIN{1|2|3|4|5|6|7|8|9|10|11|12|13|14}?

Description: Sets/Queries the primary parameters (lower limit value, upper limit value) to BIN1 - BIN14.

To execute comparison, set ":CALCulate:COMParator:PRIMary:BIN{1..14} :STATe"additionally so as to execute bin sorting.

The unit depends on the type of primary parameters to be measured or the setting of deviation display.

Parameter: <lower limit value>{OFF|numeric value, range 0, ± (1E-16 to 9.99999E+11) }

<upper limit value>{OFF|numeric value, range 0, ± (1E-16 to 9.99999E+11) }

In either case, MAX / MIN can be used.

OFF corresponds to No Limit of panel operation.

Setting example: :CALC:COMP:PRIM:BIN1 11.2345E-06, 12.3456E-06

Sets the lower limit value of BIN1 to 11.2345E-06, and the upper limit value to 12.3456E-06.

Response: <lower limit value>, <upper limit value>{OFF| numeric value, format NR3}

Query example: :CALC:COMP:PRIM:BIN1?

Response example: +1.12345E-05, +1.23456E-05

The lower limit value of BIN1 is 1.12345E-05, and the upper limit value is 1.23456E-05.

:CALCulate:COMParator:PRIMary:BIN{1|2|3|4|5|6|7|8|9|10|11|12|13|14}:STATe {ON|OFF|1|0}

:CALCulate:COMParator:PRIMary:BIN{1|2|3|4|5|6|7|8|9|10|11|12|13|14}:STATe?

Description: Sets/Queries whether the primary parameters set to the BIN1 - BIN14 are compared or not by the comparator function.

Parameter: {ON|1} Enables sorting of that bin (that bin is used for sorting)

{OFF|0} Disables sorting of that bin (that bin is not used for sorting)

Setting example: :CALC:COMP:PRIM:BIN2:STAT ON

Sets so that BIN2 sorting is executed.

Response: {1|0}

Query example: :CALC:COMP:PRIM:BIN2:STAT?

Response example: 1 BIN2 sorting is executed.

:CALCulate:COMParator:PRIMary:NOMinal <reference value>

:CALCulate:COMParator:PRIMary:NOMinal?

Description:	Sets/Queries reference value used when primary parameter is compared to obtain deviation or deviation % by the comparator function. It is common to the reference value of deviation display.
Parameter:	<reference value> {numeric value, range 0, ± (1E-16 to 9.99999E+11) } The unit depends on the parameter. MAX / MIN can be used.
Setting example:	:CALC:COMP:PRIM:NOM 12.0000E-06 Sets reference value of primary parameters to 12.0000E-06.
Response:	<reference value> {numeric value, format NR3}
Query example:	:CALC:COMP:PRIM:NOM?
Response example:	+1.20000E-05 Reference value of primary parameters is 1.20000E-05.

:CALCulate:COMParator:SECondary:LIMit <lower limit value>, <upper limit value>

:CALCulate:COMParator:SECondary:LIMit?

Description:	Sets/Queries the lower limit value and upper limit value of secondary parameters when the secondary parameters are compared by the comparator function. To execute comparison, set ":CALCulate:COMParator:SECondary:STATE" additionally so as to execute secondary parameters comparison.
Parameter:	<lower limit value>{OFF|numeric value, range 0, ± (1E-16 to 9.99999E+11) } <upper limit value>{OFF|numeric value, range 0, ± (1E-16 to 9.99999E+11) } The unit depends on the parameter. In either case, MAX / MIN can be used.
Setting example:	:CALC:COMP:SEC:LIM OFF, 0.01 Sets the lower limit value of secondary parameters to "no limit", and the upper limit value to 0.01.
Response:	<lower limit value>, <upper limit value> {OFF| numeric value, format NR3}
Query example:	:CALC:COMP:SEC:LIM?
Response example:	OFF,+1.00000E-02 The lower limit value of secondary parameters is "no limit", and the upper limit value is 0.01.

:CALCulate:COMParator:SECondary:STATe {ON|OFF|1|0}

:CALCulate:COMParator:SECondary:STATe?

Description: Sets whether the secondary parameters are compared or not by the comparator function.

Parameter: {ON|1} Secondary parameters are compared.

{OFF|0} Secondary parameters are not compared.

Setting example: :CALC:COMP:SEC:STAT ON

Sets so that secondary parameter comparison is executed.

Response: {1|0}

Query example: :CALC:COMP:SEC:STAT?

Response example: 1

Secondary parameters are compared.

:CALCulate:COMParator[:STATe] {ON|OFF|1|0}

:CALCulate:COMParator[:STATe]?

Description: Sets/queries whether the comparator function (bin sorting) is used or not.

Parameter: {ON|1} Enables the comparator function.

{OFF|0} Disables the comparator function.

Setting example: :CALC:COMP ON

Sets so that the comparator function (bin sorting) is enabled.

Response: {1|0}

Query example: :CALC:COMP?

Response example: 1

The comparator function (bin sorting) is enabled.

Remarks: If the comparator function is set to ON (enabled),

- The comparison result is displayed on the panel, and also the comparison result is output to the handler interface.
- The limit comparison function for primary and secondary parameters is set to OFF (disabled).

If the comparator function is set to OFF (disabled),

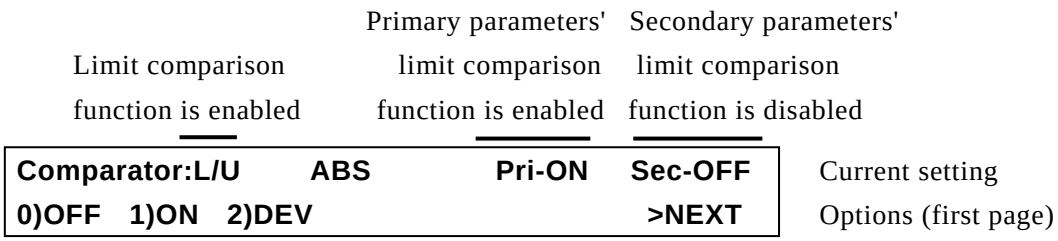
- The limit comparison function for primary and secondary parameters is also set to OFF (disabled).

Besides the bin sorting, the following limit comparison can be used.

■ Limit comparison function

In the remote control, a set of numeric value range is specified respectively for primary parameters and secondary parameters, so that they can be used for limit comparison function. If either limit comparison function of primary parameters or secondary parameters is enabled, the comparator operation changes from bin sorting to limit comparison. The limit comparison function cannot be enabled by the panel operation, except with the multi-measurement function. When the limit comparison function is enabled,

- The COMPRTR lamp on the front panel lights up.
- A set of upper and lower limit values is used for screening of primary parameters. The values are common to BIN1 of bin sorting.
- A set of upper and lower limit values is used for screening of secondary parameters. The values are common to bin sorting.
- When the display format of primary and secondary parameters is deviation or deviation %, the comparison format for limit comparison is also deviation or deviation %.
- BIN2 - BIN14 sorting is not executed. Setting to BIN1 - BIN14 is possible.
- The limit comparison is performed regardless of the settings of comparator function (bin sorting).
- The display of measured value on the front panel is same as when the comparator function (bin sorting) is enabled.
- The setting of limit comparison is displayed in the comparator setting menu (see below for instance).



The primary parameter limit comparison fail flag which is a response of :CALCulate1:LIMit:FAIL? query is updated when a new measured value is obtained. At this time, it is assumed as pass if the limit comparison function of primary parameters is set to OFF (disabled). This is also true for the secondary parameters. The limit comparison fail flag is initialized to “0” (pass) at power-on. It can also be initialized by the *RST command, initialize operation with the initialize menu or system setting menu, or :CALCulate{1|2}:LIMit:CLEar command.

:CALCulate:FORMat:AUTO[:STATe] {ON|OFF|1|0}

:CALCulate:FORMat:AUTO[:STATe]?

Description: Sets/Queries whether primary and secondary measurement parameters are selected automatically.

Parameter: {ON|1} Enables automatic selection of measurement parameters.
{OFF|0} Disables automatic selection of measurement parameters.

Setting example: :CALC:FORM:AUTO ON
Sets to enables the automatic selection of measurement parameter.

Response: {1|0}

Query example: :CALC:FORM:AUTO?

Response example: 1
The automatic selection of parameter is enabled.

Remarks: Automatic selection of measurement parameters is automatically disabled in the following cases:

- When a primary parameter is set
- When a secondary parameter is set
- When automatic selection of equivalent circuit is set to OFF (disabled)
- When a setting is made with [:SENSe]:FUNCtion[:ON] command

:CALCulate1:CKIT:AUTO[:STATe] {ON|OFF|1|0}

:CALCulate1:CKIT:AUTO[:STATe]?

Description: Sets/Queries whether equivalent circuit is selected automatically when the primary parameter is set to either R, C, or L.

Parameter: {ON|1} Enables automatic selection of equivalent circuit.
{OFF|0} Disables automatic selection of equivalent circuit.

Setting example: :CALC1:CKIT:AUTO ON
Sets to enables the automatic selection of equivalent circuit.

Response: {1|0}

Query example: :CALC1:CKIT:AUTO?

Response example: 1
The automatic selection of equivalent circuit is enabled.

:CALCulate1:FORMat {Z|Y|R|RP|RS|G|C|CP|CS|L|LP|LS|REAL|MLINear}

:CALCulate1:FORMat?

Description: Sets/Queries the primary parameter (and equivalent circuit) to be measured.

Parameter:

Z	Magnitude of impedance |Z| (unit: Ω ohm)
Y	Magnitude of admittance |Y| (unit: S siemens)
R	Resistance Rp or Rs (unit: Ω)
RP	Resistance Rp expressed with parallel equivalent circuit (unit: Ω)
RS	Resistance Rs expressed with series equivalent circuit (unit: Ω)
G	Conductance indicated with parallel equivalent circuit Gp (unit: S)
C	Capacitance Cp or Cs (unit: F farad)
CP	Capacitance Cp expressed with parallel equivalent circuit (unit: F)
CS	Capacitance Cs expressed with series equivalent circuit (unit: F)
L	Inductance Lp or Ls (unit: H henry)
LP	Inductance Lp expressed with parallel equivalent circuit (unit: H)
LS	Inductance Ls expressed with series equivalent circuit (unit: H)
REAL	Real part of immittance (either Rs or Gp)
MLINear	Magnitude of immittance (|Z|or |Y|)

Note: Immittance is generic name of impedance and admittance.

Setting example: :CALC1:FORM CS

Sets the primary parameter to the capacitance expressed with series equivalent circuit.

Response: {Z|Y|R|RP|RS|G|C|CP|CS|L|LP|LS|REAL|MLIN}

Query example: :CALC1:FORM?

Response example: CS

The primary parameter is the capacitance expressed with series equivalent circuit.

Remarks: Equivalent circuit assumed in the measurement of primary parameters is as follows:

Z, Y	Not depend on the equivalent circuit.
RS, CS, LS	Series equivalent circuit
RP, CP, LP, G	Parallel equivalent circuit

These parameters do not depend on [:SENSe]:FUNCtion[:ON] command and equivalent circuit setting.

REAL, MLINear, R, C, L

These parameters depend on [:SENSe]:FUNCtion[:ON] command and equivalent circuit setting. ☞ Below.

<When automatic selection of equivalent circuit is disabled, (:CALC1:CKIT:AUTO OFF)>

<When automatic selection of equivalent circuit is enabled, (:CALC1:CKIT:AUTO ON)>

For R, C, and L, the equivalent circuit automatic selecting function has priority over the setting of [:SENSe]:FUNCtion[:ON].

[:SENSe]:FUNCtion[:ON]	REAL	MLINear	R	C	L
FIMPedance (Series equivalent circuit)	Rs	|Z|	Rs	Cs	Ls
FADMittance (Parallel equivalent circuit)	G	|Y|	Rp	Cp	Lp
FIMPedance,FRESistance	Rs	|Z|	Rs	Cs	Ls
FADMittance,FRESistance	Rp	|Y|	Rp	Cp	Lp

:CALCulate1:LIMit:CLEar

Description: Clears the primary parameter limit comparison fail flag to “0” (pass).

Setting example: :CALC1:LIM:CLE

Clears the primary parameter limit comparison fail flag.

:CALCulate1:LIMit:FAIL?

Description: Queries whether the primary parameter limit comparison result is “fail” or not.

Response: {1|0}

1: Fail (LO or HI)

0: Pass (IN)

Query example: :CALC1:LIM:FAIL?

Response example: 0

The primary parameter is “pass”.

------ Upper and lower limit values of primary parameters in limit comparison ------

The upper and lower limit values of the primary parameters used for the limit comparison function are common to those of BIN1 for the comparator function (bin sorting). Either setting has same effect.

:CALCulate1:LIMit:LOWer[:DATA] <lower limit value>

:CALCulate1:LIMit:LOWer[:DATA]?

Description: Sets/Queries the lower limit value of primary parameters used for limit comparison.

Parameter: <lower limit value> {numeric value, range 0, ± (1E-16 to 9.99999E+11)}

MAX / MIN can be used.

Setting example: :CALC1:LIM:LOW 1.23456E-06

Sets the lower limit value of primary parameters used for limit comparison to 1.23456 µ.

Response: <lower limit value> {numeric value, format NR3}

Query example: :CALC1:LIM:LOW?

Response example: +1.23456E-06

The lower limit value of primary parameters used for limit comparison is 1.23456 µ.

:CALCulate1:LIMit:LOWer:STATe {ON|OFF|1|0}

:CALCulate1:LIMit:LOWer:STATe?

Description: Sets/Queries whether the lower limit value is compared or not in the limit comparison of primary parameters.

Parameter: {ON|1} Lower limit value is compared.

{OFF|0} Lower limit value is not compared

(corresponding to No Limit of panel operation).

Setting example: :CALC1:LIM:LOW:STAT ON

Sets so that the lower limit value is compared in the primary parameter limit comparison.

Response: {1|0}

Query example: :CALC1:LIM:LOW:STAT?

Response example: 1

The lower limit value is compared in the primary parameter limit comparison.

:CALCulate1:LIMit:STATe {ON|OFF|1|0}

:CALCulate1:LIMit:STATe?

Description: Sets/Queries whether the limit comparison function of primary parameter is used or not.

Parameter: {ON|1} Enables the limit comparison function of primary parameter.

{OFF|0} Disables the limit comparison function of primary parameter.

Setting example: :CALC1:LIM:STAT ON

Sets to enables the limit comparison function of primary parameter.

Response: {1|0}

Query example: :CALC1:LIM:STAT?

Response example: 1

The limit comparison function of primary parameter is enabled.

Remarks: If the primary parameter limit comparison function is enabled.

- The comparator operation changes to the limit comparison.
- BIN2 - BIN14 sorting is not executed. (Bin sorting is restricted)
- The comparison result is displayed on the panel.
- The comparison result is output to the handler interface.

If the primary parameter limit comparison function is disabled, the comparator function is set to OFF (disabled) if the secondary parameter limit comparison is disabled.

:CALCulate1:LIMit:UPPer[:DATA] < upper limit value>

:CALCulate1:LIMit:UPPer[:DATA]?

Description: Sets/Queries the upper limit value of primary parameters used for limit comparison.

Parameter: <upper limit value>{numeric value, range 0, ± (1E-16 to 9.99999E+11)}
MAX / MIN can be used.

Setting example: :CALC1:LIM:UPP 12.3456E-06
Sets the upper limit value of primary parameters used for limit comparison to 12.3456 µ.

Response: <upper limit value>{numeric value, format NR3}

Query example: :CALC1:LIM:UPP?

Response example: +1.23456E-05
The upper limit value of primary parameters used for limit comparison is 1.23456E-05 (=12.3456 µ).

:CALCulate1:LIMit:UPPer:STATe {ON|OFF|1|0}

:CALCulate1:LIMit:UPPer:STATe?

Description: Sets/Queries whether the upper limit value is compared or not in the limit comparison of primary parameters.

Parameter: {ON|1} Upper limit value is compared.
{OFF|0} Upper limit value is not compared (corresponding to No Limit of panel operation).

Setting example: :CALC1:LIM:UPP:STAT ON
Sets so that the upper limit value is compared in the primary parameter limit comparison.

Response: {1|0}

Query example: :CALC1:LIM:UPP:STAT?

Response example: 1
The upper limit value is compared in the primary parameter limit comparison.

:CALCulate1:MATH:EXPRession:NAME {DEV|PCNT}

:CALCulate1:MATH:EXPRession:NAME?

Description: Sets/Queries the deviation display format of primary parameters.
To display the deviation, set the deviation display format, and further set so that the deviation is displayed using :CALCulate1:MATH:STATe ON command.
Set the referenced value with :DATA[:DATA] command.

Parameter: DEV Deviation: Deviation from reference value
(= measured value – reference value) is displayed.
PCNT Deviation%: Displays the deviation% (from the reference value).

Setting example: :CALC1:MATH:EXPR:NAME PCNT
Sets the deviation display format of primary parameters to deviation %.

Response: {DEV|PCNT}

Query example: :CALC1:MATH:EXPR:NAME?

Response example: PCNT
The deviation display format of primary parameters is deviation %.

:CALCulate1:MATH:STATe {ON|OFF|1|0}

:CALCulate1:MATH:STATe?

Description: Sets/Queries whether the primary parameter deviation display / output is executed or not.

Parameter: {ON|1} Enables the deviation display / output.
{OFF|0} Disables the deviation display / output.

Setting example: :CALC1:MATH:STAT ON
Sets to enables the primary parameter deviation display / output.

Response: {1|0}

Query example: :CALC1:MATH:STAT?

Response example: 1
The primary parameter deviation display / output is enabled.

Remarks: If the primary parameters or secondary parameters are changed with :CALCulate{1|2}:FORMat command, the deviation display / output of both primary and secondary parameters is disabled automatically. Set the primary parameters and secondary parameters prior to the deviation display / output.

:CALCulate2:FORMat {Q|D|PHASe|X|B|RS|RP|G|LP|RDC|IMAGinary|REAL}
:CALCulate2:FORMat?

Description: Sets the secondary parameters to be measured.

Parameter:

Q	Quarity factor (=1/D)
D	Dissipation factor (= tanδ)
PHASe	Phase angle of impedance (θ, unit: degree)
X	Reactance Xs expressed with series equivalent circuit (unit: Ω)
B	Susceptance Bp expressed with parallel equivalent circuit (unit: S)
RS	Resistance Rs expressed with series equivalent circuit (unit: Ω)
RP	Resistance Rp expressed with parallel equivalent circuit (unit: Ω)
G	Conductance Gp expressed with parallel equivalent circuit (unit: S)
LP	Inductance Lp expressed with parallel equivalent circuit (unit: H)
RDC	Direct-current resistance Rdc (unit: Ω)
IMAGinary	Imaginary part of immittance (X or B)
REAL	Real part of immittance (either Rs, Gp or Rdc)
	Note: Immittance is generic name of impedance and admittance.

Setting example: :CALC2:FORM D
Sets the secondary parameter to D.

Response: {Q|D|PHAS|X|B|RS|RP|G|LP|RDC|IMAG|REAL}

Query example: :CALC2:FORM?

Response example: D
The secondary parameter is dissipation factor D.

Remarks: Equivalent circuit assumed in the measurement of secondary parameters is as follows:

PHASe, D, Q, RDC	Not depend on the equivalent circuit.
X, RS	Series equivalent circuit
G, B, LP, RP	Parallel equivalent circuit

These parameters do not depend on [:SENSe]:FUNCtion[:ON] command and equivalent circuit setting.

IMAGinary, REAL

These parameters depend on [:SENSe]:FUNCtion[:ON] command and equivalent circuit setting. ☞ Table below

[:SENSe]:FUNCtion[:ON]	IMAGinary	REAL
FIMPedance (Series equivalent circuit)	X	Rs (=ESR)
FADMittance (Parallel equivalent circuit)	B	G
FIMPedance,FRESistance	X	Rdc
FADMittance,FRESistance	B	Rdc

The setting of equivalent circuit (series/parallel) and the setting of automatic selection of equivalent circuit do not affect the secondary parameter setting. However, the secondary parameters actually measured and displayed may vary since [:SENSe]:FUNCtion[:ON] command and equivalent circuit setting are interlocked.

:CALCulate2:LIMit:CLEar

Description: Clears the secondary parameter limit comparison fail flag to “0” (pass).
Setting example: :CALC2:LIM:CLE
Clears the secondary parameter limit comparison fail flag.

:CALCulate2:LIMit:FAIL?

Description: Queries whether the secondary parameter limit comparison result is “fail” or not.
Response: {1|0}
1: Fail (LO or HI)
0: Pass (IN)
Query example: :CALC2:LIM:FAIL?
Response example: 0
The secondary parameter is “pass”.

------ Upper and lower limit values of secondary parameters in limit comparison ------

The upper and lower limit values used for the limit comparison function are common to those of secondary parameter for the comparator function (bin sorting). Either setting has same effect.

:CALCulate2:LIMit:LOWer[:DATA] <lower limit value>
:CALCulate2:LIMit:LOWer[:DATA]?

Description: Sets/Queries the lower limit value of secondary parameters used for limit comparison.
Parameter: <lower limit value>{numeric value, range 0, ± (1E-16 to 9.99999E+11)}
MAX / MIN can be used.
Setting example: :CALC2:LIM:LOW 1.23456E-06
Sets the lower limit value of secondary parameters used for limit comparison to 1.23456 µ.
Response: <lower limit value>{numeric value, format NR3}
Query example: :CALC2:LIM:LOW?
Response example: +1.23456E-06
The lower limit value of secondary parameters used for limit comparison is 1.23456 µ.

:CALCulate2:LIMit:LOWer:STATe {ON|OFF|1|0}

:CALCulate2:LIMit:LOWer:STATe?

Description: Sets/Queries whether the lower limit value is compared or not in the limit comparison of secondary parameters.

Parameter: {ON|1} Lower limit value is compared.

{OFF|0} Lower limit value is not compared (corresponding to No Limit of panel operation).

Setting example: :CALC2:LIM:LOW:STAT ON

Sets so that the lower limit value is compared in the secondary parameter limit comparison.

Response: {1|0}

Query example: :CALC2:LIM:LOW:STAT?

Response example: 1

The lower limit value is compared in the secondary parameter limit comparison.

:CALCulate2:LIMit:STATe {ON|OFF|1|0}

:CALCulate2:LIMit:STATe?

Description: Sets/Queries whether the limit comparison function of secondary parameter is used or not.

Parameter: {ON|1} Enables the limit comparison function of secondary parameter.

{OFF|0} Disables the limit comparison function of secondary parameter.

Setting example: :CALC2:LIM:STAT ON

Sets to enables the limit comparison function of secondary parameter.

Response: {1|0}

Query example: :CALC2:LIM:STAT?

Response example: 1

The limit comparison function of secondary parameter is enabled.

Remarks: If the secondary parameter limit comparison function is enabled,

- The comparator operation changes to the limit comparison.
- BIN2 - BIN14 sorting is not executed. (Bin sorting is restricted)
- The comparison result is displayed on the panel.
- The comparison result is output to the handler interface.

If the secondary parameter limit comparison function is disabled, the comparator function is set to OFF (disabled) if the primary parameter limit comparison is disabled.

:CALCulate2:LIMit:UPPer[:DATA] <upper limit value>

:CALCulate2:LIMit:UPPer[:DATA]?

Description: Sets/Queries the upper limit value of secondary parameters used for limit comparison.

Parameter: <upper limit value>{numeric value, range 0, ± (1E-16 to 9.99999E+11)}
MAX / MIN can be used.

Setting example: :CALC2:LIM:UPP 12.3456E-06
Sets the upper limit value of secondary parameters used for limit comparison to 12.3456 µ.

Response: <upper limit value>{numeric value, format NR3}

Query example: :CALC2:LIM:UPP?

Response example: +1.23456E-05
The upper limit value of secondary parameters used for limit comparison is 1.23456E-05 (= 12.3456 µ).

:CALCulate2:LIMit:UPPer:STATe {ON|OFF|1|0}

:CALCulate2:LIMit:UPPer:STATe?

Description: Sets/Queries whether the upper limit value is compared or not in the limit comparison of secondary parameters.

Parameter: {ON|1} Upper limit value is compared.
{OFF|0} Upper limit value is not compared
(corresponding to No Limit of panel operation).

Setting example: :CALC2:LIM:UPP:STAT ON
Sets so that the upper limit value is compared in the secondary parameter limit comparison.

Response: {1|0}

Query example: :CALC2:LIM:UPP:STAT?

Response example: 1
The upper limit value is compared in the secondary parameter limit comparison.

:CALCulate2:MATH:EXPRession:NAME {DEV|PCNT}

:CALCulate2:MATH:EXPRession:NAME?

Description: Sets/Queries the deviation display format of secondary parameters.
To display the deviation, set the deviation display format, and further set so that the deviation is displayed using :CALCulate2:MATH :STATe ON command.
Set the referenced value with :DATA[:DATA] command.

Parameter: DEV Deviation: Deviation from reference value (= measured value – reference value) is displayed.

PCNT Deviation%: Displays the deviation% (from the reference value).

Setting example: :CALC2:MATH:EXPR:NAME DEV
Sets to deviation the deviation display format of secondary parameters.

Response: {DEV|PCNT}

Query example: :CALC2:MATH:EXPR:NAME?

Response example: DEV
The deviation display format of secondary parameters is deviation.

:CALCulate2:MATH:STATe {ON|OFF|1|0}

:CALCulate2:MATH:STATe?

Description: Sets/Queries whether the secondary parameter deviation display / output is executed or not.

Parameter: {ON|1} Enables the deviation display / output.
{OFF|0} Disables the deviation display / output.

Setting example: :CALC2:MATH:STAT OFF
Sets to disables the secondary parameter deviation display / output.

Response: {1|0}

Query example: :CALC2:MATH:STAT?

Response example: 0
The secondary parameter deviation display / output is disabled.

:CALCulate3:MATH:STATe {ON|OFF|1|0}

Description: Sets whether the current monitor is displayed or not.

Parameter: {ON|1} Sets auxiliary display to the voltage & current monitor.
This setting is equivalent to :DISP:TEXT3 MON.
{OFF|0} Stops the display on the voltage & current monitor.
Only when :DISP:TEXT3 is set to MON, set to :DISP:TEXT3 STAT.
No influence for others.

Setting example: :CALC3:MATH:STAT ON
Sets auxiliary display to the voltage & current monitor.

:CALCulate4:MATH:STATe {ON|OFF|1|0}

Description:	Sets whether the voltage monitor is displayed or not.	
Parameter:	{ON|1}	Sets auxiliary display to the voltage & current monitor.
		This setting is equivalent to :DISP:TEXT3 MON.
	{Off|0}	Stops the display on the voltage & current monitor.
		Only when :DISP:TEXT3 is set to MON,
		set to :DISP:TEXT3 STAT.
		No influence for others.
Setting example:	:CALC4:MATH:STAT ON	
	Sets auxiliary display to the voltage & current monitor.	

:CALibration:CABLe <cable length>

:CALibration:CABLe?

Description:	Sets/Queries the measurement cable length.
Parameter:	<cable length> {numeric value, range 0, 1, 2, 4, unit: m}
	A numeric value is rounded to 0, 1, 2, or 4, whichever nearest.
Setting example:	:CAL:CABL 1
	Sets the cable length to 1 m.
Response:	<cable length> {numeric value, format NR1}
Query example:	:CAL:CABL?
Response example:	+1
	The cable length is 1 m.

:DATA[:DATA] {REF1|REF2}, <reference value>

:DATA[:DATA]? {REF1|REF2|BUF1|BUF2|BUF3|IMON|VMON|VSOU}

Description: Sets reference value used when the deviation is displayed. It is common to the reference value of comparator.
Or, queries reference value when the deviation is displayed, or measured data.

Parameter:

REF1	Sets/Queries reference value for deviation display/comparison of primary parameters.
REF2	Sets/Queries reference value for deviation display/comparison of secondary parameters.
BUF1	Queries the measured data buffer 1.
BUF2	Queries the measured data buffer 2.
BUF3	Queries the measured data buffer 3.
IMON	Queries the current monitored value.
VMON	Queries the voltage monitored value.
VSOU adjusted	Queries the measurement voltage level automatically by ALC.
<reference value>	REF1 {NR3, range 0, ± (1E-16 to 9.99999E+11)} REF2 {NR3, range 0, ± (1E-16 to 9.99999E+11)} In either case, MAX / MIN can be used.

The monitored value can be queried even if the monitored value is not displayed on the auxiliary display section.

Setting example: :DATA REF1, 1.23456E-06
Sets reference value of primary parameters to 1.23456E-06.

Response:

REF1	{NR3}
REF2	{NR3}
IMON	{NR3, range 0, 1E-10 to 9.99999E-01, unit: Arms}
VMON	{NR3, range 0, 1E-05 to 9.99999E+00, unit: Vrms} Current and voltage monitored values are measured values. Normally, the output values are up to about 200 mArms and 5 Vrms.
VSOU	{NR3, range 0, 1E-02 to 5.0000E+00, unit: Vrms} If ALC is disabled, the measurement voltage level setting value is returned.

BUF1, BUF2, BUF3
<measured data 1>,<measured data 2>,<measured data 3>, .., <measured data n>

"n" is the measurement point number specified by :DATA:POINts command.
The read data is deleted from the measured data buffer.
The parts not recorded are all measured data zero (for example, +0,+0.00000E+00,+0).

The content of <measured data m> is as follows (m: 1 to n).

<measurement status>, <measured data> [, {<comparator bin sorting result> | <limit comparison result> }]

<measurement status>{numeric value, format NR1, range see below}

0 No error

1 Either measurement error ERR, ALC error ALC Err, or correction error CORR Err

2 Contact failure NC, or abnormally low capacitance LoC

3 Any other errors

<measured value>

For BUF1 and BUF2, either <primary parameter measured value> or <secondary parameter measured value> set by :DATA:FEED command.

For BUF3, <primary parameter measured value> and <secondary parameter measured value>.

For data transfer format, ☞ :FORMat[:DATA] command

<comparator's bin sorting result> {numeric value, format NR1, range see below}

For BUF1 and BUF2, the bin sorting result is output whichever the comparator function (bin sorting) is enabled or disabled.

For BUF3, the bin sorting result is output only when the comparator function (bin sorting) is enabled.

When bin extended function is disabled		When bin extended function is enabled	
0	OUT_OF_BINS	0	OUT_OF_BINS
1	BIN1	1	BIN1
2	BIN2	2	BIN2
	••		••
9	BIN9	9	BIN9
10	AUX_BIN (S-NG)	10	BIN10
11	Sorting failed	11	BIN11
			••
		14	BIN14
		15	AUX_BIN (S-NG)
		16	Sorting failed

AUX_BIN (S-NG): Output only when S-NG is used as auxiliary bin.

☞ :CALCulate:COMParator:AUXBin command

Sorting failed: Any error occurred or comparator function is disabled.

When the bin extended function is enabled, the range exceeding 10 has different meaning.

☞ :CALCulate:COMParator:EXTension[:STATe] command

<limit comparison result>

When the primary parameter or secondary parameter limit comparison function is enabled, the limit comparison result is output regardless of the setting of comparator function (bin sorting). The bin sorting result of comparator is not output.

For BUF1 and BUF2, either <primary parameter comparison result> or <secondary parameter comparison result> set by :DATA:FEED command is output.

For BUF3, <primary parameter comparison result> and <secondary parameter comparison result> are output.

<{primary|secondary}parameter comparison result> {NR1, range see below}

0 Limit comparison function is disabled
1 IN(within the range of upper limit and lower limit)
2 HI(larger than upper limit)
4 LO(smaller than lower limit)

Query example 1: :DATA? REF1

Response example 1:+1.23456E-06

The reference value of primary parameter is 1.23456E-06.

Query example 2: :DATA? BUF1

Response example 2:Only one measured data is shown here as an example, when the comparator function is enabled.

+0,+1.23456E-06,+2

No measurement error, Measured value = 1.23456E-06, Comparator's bin sorting result = Bin 2

Query example 3: :DATA? BUF3

Response example 3:Only one measured data is shown here as an example, when the limit comparison function is enabled.

+0,+1.23456E-06,+1.43657E-03,+1,+2

No measurement error, Primary parameter 1.23456E-06, Secondary parameter 1.43657E-03, Primary parameter limit comparison IN, Secondary parameter limit comparison HI.

Remarks: For multi-measurement, measured data for each step is recorded with the limit comparison result.

:DATA:FEED {BUF1|BUF2},{"CALCulate1"|"CALCulate2"|""}

:DATA:FEED? {BUF1|BUF2}

Description: Sets/Queries the measured data recorded in the measured data buffer 1 or measured data buffer 2. To record the measured data, specify additionally the :DATA:FEED:CONT command so as to record the measured data.

Parameter:
- BUF1 Sets/Queries the measured data buffer 1.
- BUF2 Sets/Queries the measured data buffer 2.
- "CALCulate1" Records the measured data of primary parameter.
- "CALCulate2" Records the measured data of secondary parameter.
- "" Both measured data are not recorded.

Double quotation (") is required to indicate the measured data to be recorded.

Setting example: :DATA:FEED BUF1,"CALC1"
Sets so that the measured data of primary parameter is recorded in the measured data buffer 1.

Response: {"CALC1"|"CALC2"|""}

Query example: :DATA:FEED? BUF1

Response example: "CALC1"
The measured data recorded in the measured data buffer 1 is primary parameter.

Remarks: Measured values to be recorded in BUF3 are fixed to <primary parameter measured value> and <secondary parameter measured value>.
Details ☞ :DATA[:DATA] command

:DATA:FEED:CONTrol {BUF1|BUF2|BUF3},{ALWays|NEVer}

:DATA:FEED:CONTrol? {BUF1|BUF2|BUF3}

Description: Sets/Queries whether the measured data is recorded in the measured data buffer.

Parameter:
- BUF1 Sets/Queries the measured data buffer 1.
- BUF2 Sets/Queries the measured data buffer 2.
- BUF3 Sets/Queries the measured data buffer 3.
- ALWays The measured data is recorded every measurement.
- NEVer The measured data is not recorded.

Setting example: :DATA:FEED:CONT BUF1,ALW
Sets so that the measured data is recorded in the measured data buffer 1.

Response: {ALW|NEV}

Query example: :DATA:FEED:CONT? BUF1

Response example: ALW
The measured data buffer 1 is set so as to record the measured data.

Remarks: About the measured data to be recorded
Details ☞ :DATA[:DATA]? query, :DATA:FEED command
If the measured data is set to be recorded in the measured data buffer, it is not sent to the output buffer.

:DATA:POINts {BUF1|BUF2|BUF3},<buffer size>

:DATA:POINts? {BUF1|BUF2|BUF3}

Description: Sets/Queries virtual size (number of measurement points) of the measured data buffer.
If set, the specified measured data buffer is cleared.

Parameter: BUF1 Sets/Queries the measured data buffer 1.
BUF2 Sets/Queries the measured data buffer 2.
BUF3 Sets/Queries the measured data buffer 3.
<buffer size> Expressed with the number of measurement points.
MAX / MIN can be used.
BUF1, BUF2 {numeric value, range 1 to 200}
BUF3 {numeric value, range 1 to 1000}

Setting example: :DATA:POIN BUF1,100
Sets virtual size of measured data buffer 1 to 100.

Response: BUF1, BUF2 {numeric value, format NR1}
BUF3 {numeric value, format NR1}

Query example: :DATA:POIN? BUF1

Response example: +100
Virtual size of measured data buffer 1 is 100.

Remarks: The measured data buffer is a ring buffer. After it becomes physically full, the data in the lead position is overwritten. When the measured data are recorded by the amount of virtual size, the status that indicates “full” is set.
••• ☞ “6.4 Operation Status”
For :DATA[:DATA]? {BUF1|BUF2|BUF3}, the data by the amount of virtual size are transmitted starting from the lead position, whichever the buffer is full or not.

:DISPlay[:WINDow][:STATe] <display permission>

Description: This command is accepted but it does nothing.

Parameter: <display permission> {ON|OFF|1|0}

Setting example: :DISP OFF
Nothing is done. The measured result is always displayed.

:DISPlay[:WINDow]:TEXT1:DIGit <display digits>

Description: This command is accepted but it does nothing.

Parameter: <display digits> {numeric value, range 3 - 6}
Out of range value causes an error.

Setting example: :DISP:TEXT1:DIG 4
Nothing is done. The measured value is always displayed with the maximum of digits.

:DISPlay[:WINDow]:TEXT1:PAGE <displayed data>

Description: This command is accepted but it does nothing.

Parameter: <displayed data> {numeric value, range 1 or 2}
Out of range value causes an error.

Setting example: :DISP:TEXT1:PAGE 2
Nothing is done.
The comparator result and limit comparison result are displayed simultaneously with the measured value.

:DISPlay[:WINDow]:TEXT2:PAGE <display item>

Description: Sets auxiliary display items. Converted into :DISP:TEXT3 command and executed.

Parameter: <display item> {numeric value, range 1 - 8}
Out of range value causes an error.

2	Measurement condition status
4	Lower limit value and upper limit value of BIN1 (primary parameter)
5	Lower and upper limit values of secondary parameter
7	Voltage & current monitor
1, 3, 6, 8	Content of auxiliary display is not changed but previous display item is retained.

Setting example: :DISP:TEXT2:PAGE 4
Sets so as to display the lower limit value and upper limit value of BIN1 (primary parameter).

:DISPlay[:WINDow]:TEXT3[:PAGE] {STATus|BIN1|..|BIN14|SLIMit|REFerence|MONitor|IMPedance|LIST}

:DISPlay[:WINDow]:TEXT3[:PAGE]?

Description:	Sets/Queries auxiliary display item.	
Parameter:	STATus	Measurement condition status
	BIN1..BIN14	Lower limit value and upper limit value of the specified bin of primary parameter
	SLIMit	Lower limit value and upper limit value of secondary parameter (Secondary LIMit)
	REFerence	Reference values of primary parameter and secondary parameter (for deviation)
	MONitor	Current and voltage monitored values (I-V)
	IMPedance	Impedance measured value (Z-θ)
	LIST	Steps during multi-measurement
Setting example:	:DISP:TEXT3 MON	
	Sets auxiliary display item to the voltage & current monitor.	
Response:	{STAT|BIN1|..|BIN14|SLIM|REF|MON|IMP|LIST}	
Query example:	:DISP:TEXT3?	
Response example:	MON	
	Auxiliary display item is the voltage & current monitor.	
Remarks:	To display the lower limit value and upper limit value of primary parameter used for the limit comparison function, specify the BIN1. Do not specify a bin by mistake which is not used for the limit comparison.	

:FETCh?

Description: Queries the latest measured data.
For the data transfer format, see :FORMat[:DATA] command.

Response: When comparator function and limit comparison function are both disabled
<measurement status>,<primary parameter measured value>, <secondary parameter measured value>
When comparator function (bin sorting) is enabled
<measurement status>,<primary parameter measured value>, <secondary parameter measured value>,
<comparator's bin sorting result>
When limit comparison function is enabled
<measurement status>,<primary parameter measured value>, <secondary parameter measured value>,
<limit comparison result>
The limit comparison result is output regardless of the setting of comparator function (bin sorting). The bin sorting result of comparator is not output.

The contents of respective data are as follows.
<measurement status>
0 No error
1 Either measurement error ERR, ALC error ALC Err, or correction error CORR Err
2 Contact failure NC, or abnormally low capacitance LoC
3 Any other errors

<primary parameter measured value> and <secondary parameter measured value>
{NR3, range ±9.99999E+11}
<comparator's bin sorting result>

When bin extended function is disabled

0	OUT_OF_BINS
1	BIN1
2	BIN2
	• •
9	BIN9
10	AUX_BIN (S-NG)
11	Sorting failed

When bin extended function is enabled

0	OUT_OF_BINS
1	BIN1
2	BIN2
	• •
9	BIN9
10	BIN10
11	BIN11
	• •
14	BIN14
15	AUX_BIN (S-NG)
16	Sorting failed

AUX_BIN (S-NG): Output only when S-NG is used as auxiliary bin.

☞ :CALCulate:COMParator:AUXBin command

Sorting failed: Any error occurred or comparator function is disabled.

When the bin extended function is enabled, the range exceeding 10 has different meaning.

☞ :CALCulate:COMParator:EXTension[:STATe] command

<limit comparison result> [,<primary parameter comparison result>] [,<secondary parameter comparison result>]

Only the comparison result of the parameter that enabled the limit comparison function is output.

0 Limit comparison function is disabled
1 IN(within the range of upper limit and lower limit)
2 HI(larger than upper limit)
4 LO(smaller than lower limit)

Query example: :FETC?

Response example: +0,+3.14159E-06,+1.20000E-02,+2

(When comparator function enabled, C-D measurement, deviation not displayed, ASCII format)

Measurement status = 0 (no error), C = 3.14159E-06 (3.14159 µF),
D = 1.20000E-02 (0.01200), Comparator's bin sorting result = 2 (BIN2)

Remarks: When the measurement status is other than 0 (error), the measured value of primary parameter and secondary parameter is 9.9E+37, and the comparator result is 11 (16 when bin extended function is enabled). When the measurement status is 1 or 3, the limit comparison result is 2 (HI).
[The abnormally low capacity LoC is detected, measurement status becomes the value 2 which shows poor contact, and measured value is outputted.]

:FORMat[:DATA] {ASCii|REAL[,64]|PACKed}

:FORMat[:DATA]?

Description:	Sets/Queries the data transfer format.	
	The set data transfer format is applied to the response message of :DATA[:DATA]?, :FETCh?, or :READ? query.	
Parameter:	ASCii	ASCII format (NR1/NR2/NR3, character string)
	REAL[,64]	Real number format (double-precision floating point, 64-bit binary)
	PACKed	Packed format (display digit fixed ASCII format)
Setting example:	:FORM ASC	
	Sets the data transfer format to ASCII.	
Response:	{ASC|REAL|PACK}	
Query example:	:FORM?	
Response example:	ASC	
	The data transfer format is ASCII.	

■ ASCII format

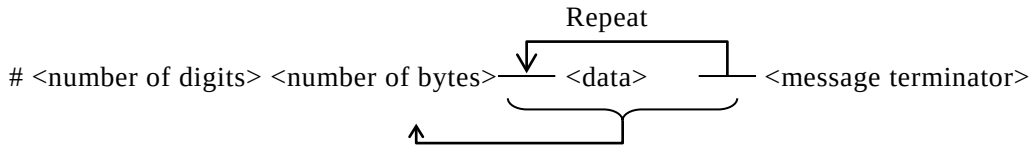
The data is expressed with characters, and respective characters are transferred with ASCII codes. Depending on the parameter, a numeric value is transferred in either of the following formats:

NR1	Integer (ex. : +123, measurement status or comparison result)
NR2	Explicit decimal point format not having exponent (ex. : +0.12345)
NR3	Explicit decimal point format having exponent (ex. : +1.23456E-07, primary parameter measured value, etc.)

When multiple numeric values are transferred, numeric values are separated with a comma (,). At the end, a message terminator (for instance, LF^EOI) is added.

■ Real number format

The data is transferred in the 64-bit double-precision floating point format in accordance with IEEE 754. When multiple numeric values are transferred, all data are expressed in the double-precision floating point format.

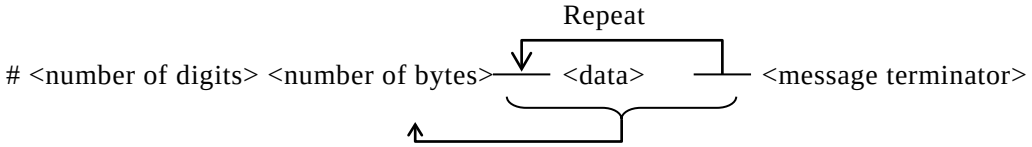


#	# is attached at the leading position.	ASCII
<number of digits>	Number of digits of character string that indicates the number of bytes (1 byte).	
<number of bytes>	Number of bytes of all data (multiples of 8).	
<data>	Double-precision floating point (8 bytes). Sign 1 bit, exponential part 11 bits, decimal part 52 bits from higher-order bit. Data are transferred from most significant byte toward lower bytes in order.	Binary

When multiple numeric data are transferred, they are fed successively without separating them with a comma (,).
At the end, a message terminator is added. Note that a message terminator is added at the end, different from the definite length arbitrary data block specified by the IEEE 488.2 Standard.

■ Packed format

The measured value is transferred in the ASCII character string format in which the display digits are fixed.
Numeric value is transferred with definite length ASCII character string in which an exponential part and decimal point position are fixed.



#	# is attached at the leading position.	ASCII
<number of digits>	Number of digits of character string that indicates the number of bytes. (1 byte)	
<number of bytes>	Number of bytes of all data.	
<data>	ASCII character string of number of characters specified every type of parameters.	

When multiple numeric data are transferred, they are fed successively without separating them with a comma (,).
At the end, a message terminator (LF^EOI) is added. Note that a message terminator (LF^EOI) is added at the end, different from the definite length arbitrary data block specified by the IEEE 488.2 Standard.

Each <data> is expressed as follows:

<measurement status>, <primary parameter comparison result>, <secondary parameter comparison result>

Number of characters 1

Content 1-digit integer indicating respective results 0 to 8.

<comparator's bin sorting result>

Number of characters2

Content 2-digit integer indicating <comparator's bin sorting result> 00 to 16.

<primary parameter measured value>, <secondary parameter measured value>

Number of characters10

Content The following three items are successively placed without separating them with a comma (,), space, or decimal point.

Sign (1 character {+|-})

Count value indicating the magnitude of measured value

(6 characters 000000 to 999999)

Exponential part (3 characters -99 to +00 to +99)

Measured value = Sign factor {+1|-1} × Count value × (“exponential part” power of 10)

Specifying the digit

For the display digits (decimal point position and exponential part) of data such as L, C and R, specify the reference value for deviation measurement. For example, if the reference value for deviation measurement is set to 12 m (12.0000 m), the measured value 00.0000 m to 99.9999 m is converted into count value 000000 to 999999 and exponential part -07, and output. The deviation is also converted similarly.

Deviation% ±999.999 % is output as follows:

sign {+|-}, count value 000000 to 999999, exponential part -03

Phase θ ±180.000 is output as follows:

sign {+|-}, count value 000000 to 180000, exponential part -03

The count value exceeding 999999 is all converted into 999999. At this time, the sign is retained.

:INITiate:CONTinuous {ON|OFF|1|0}

:INITiate:CONTinuous?

Description: Sets/Queries whether the trigger system continuous initiation is enabled or disabled.

If enabled, when the measurement finished and the trigger system becomes idle state, the instrument goes in the trigger waiting state automatically. If disabled, the instrument remains in idle state.

Setting example: :INIT:CONT ON

Enables the trigger system continuous initiation.

Response: {1|0}

Query example: :INIT:CONT?

Response example: 1

The trigger system continuous initiation is enabled.

Remarks: The continuous initiation is enabled at power-on.

It is also enabled when the initialization is executed by panel operation.

The continuous initiation is disabled by the *RST command.

:INITiate[:IMMediate]

Description: The instrument goes in the trigger waiting state when the trigger system is in idle state.

Setting example: :INIT

Sets the trigger waiting state.

:LIST:FREQuency <Frequency 1>, <Frequency 2>, .. , <Frequency n>
:LIST:FREQuency?

Description: Sets/Queries the frequency registered in the multi-measurement list. In multi-measurement, only the steps with a frequency registered are measured. The other steps are disabled and not measured.

Parameter: Specifies the measurement frequency <Frequency m>{numeric value, range 0.02 to 5.5E6 or 9.9E37}, maximum 32 points.
Initially, all steps are disabled (9.9E37).
A step with an invalid frequency (9.9E37) and subsequent steps are not measured.

Setting example: :LIST:FREQ 1E3, 100E3, 1E6
Registers the frequency 1 kHz, 100 kHz, and 1 MHz in the multi-measurement list in this order.

Response: <Frequency 1>, <Frequency 2>, .. , <Frequency 32>
<Frequency m>{NR3}

Query example: :LIST:FREQ?

Response example: 1.00000E+03, 100.000E+03, 1.00000E+06, 9.90000E+37,•••, 9.90000E+37
Frequencies registered in the multi-measurement list are 1 kHz, 100 kHz, and 1 MHz in this order. All the remaining steps are disabled (not measured).

:LIST:MEMory <Memory 1>, <Memory 2>, .. , <Memory n>
:LIST:MEMory?

Description: Sets/queries the setting memory registered in the multi-measurement list. Main parameters other than the frequency are recalled from the setting memory in each step. Below the maximum number of points, all the remaining memory specifications are disabled.

Parameter: Specifies memory <Memory m> {numeric value, range 0 to 31 or 999}, maximum 32 points.
Initially, memory specifications are disabled (999) in all steps.
If a memory specification is disabled or has the same setting memory number as the previous step, the previous setting is held without recalling the setting in this step.

Setting example: :LIST:MEM 0, 1, 1, 2
Registers the 0, 1, 1, and 2 in the multi-measurement list in this order.

Response: <Memory 1>, <Memory 2>, .. , <Memory 32>
<Memory m> {numeric value, format NR1, range 0 to 31 or 999}

Query example: :LIST:MEM?

Response example: 0, 1, 1, 5, 2, 999,..., 999
Setting memory specifications registered in the multi-measurement list are 0, 1, 1 (held), 5, and 2 in this order. Memory specifications for all the remaining steps are disabled.

:LIST:MODE {SEQuence|STEPped}

:LIST:MODE?

Description: Sets/queries whether the multi-measurement mode is sequence or step. To enable the multi-measurement, use the :LIST[:STATe] ON command to set it separately.

Parameter: SEQuence Sets the sequence mode (initial value).
All points are measured in the order of the list once triggered.
STEPped Sets the step mode.
One point is measured in the order of the list at each trigger.

Setting example: :LIST:MODE SEQ
Sets the sequence mode.

Response: {SEQ|STEP}

Query example: :LIST:MODE?

Response example: SEQ
The sequence mode.

Remarks: :ABORt command can interrupt the multi-measurement.
The result of multi-measurement can be imported in the following procedure.
For the sequence mode, use the :TRIGger[:IMMediate] command to apply a trigger and use the :FETCt? query to read measured data at once from the output buffer. After the measurement is finished for all the steps, the measured data is transferred.
For the step mode, use the *TRG command to apply a trigger in each step and read the measured data in each step from the output buffer.
Alternatively, you can send measured values to the measured data buffer, confirm that the measurement is finished for all the steps, and then read the measured data at once.

:LIST[:STATe] {ON|OFF|1|0}

:LIST[:STATe]?

Description: Sets/Queries whether the multi-measurement is executed or not.

Parameter: {ON|1} Executes the multi-measurement.
{OFF|0} Executes the normal measurement (initial value).

Setting example: :LIST ON
Sets it to execute the multi-measurement.

Response: {1|0}

Query example: :LIST?

Response example: 1
It is set to execute the multi-measurement.

:READ?

Description:	The measurement is stopped and the instrument goes in the trigger waiting state whichever state the instrument is placed in at that time. After that, when the trigger is applied and one-time measurement finished, the measured data is read. After :READ? query is transmitted, receive the measured data. Upon completion of the measurement, newly measured data can be received.
Response:	<measurement status>, <primary parameter measured value>, <secondary parameter measured value> [, {<comparator bin sorting result> | <limit comparison result> }]
Query example:	:READ?
Response example:	+0,+3.1415E-06,+1.20000,+2 Response message is same as :FETCh? query. For details, see the description of :FETCh? query.
Remarks:	:READ? is almost equivalent to :ABORt; :INITiate; FETCh?. Since the :READ? query itself does not apply the trigger, the trigger is required additionally. Once the :READ? query is executed, the instrument does not execute the next command until the measured data is placed in the output buffer. For this reason, the trigger is not executed even if the trigger source is set to BUS and an attempt is made to apply the trigger by remote control immediately after the :READ? query, causing the instrument to be hung up in the trigger waiting state. The trigger waiting state can be reset by the device clear.

[:SENSe]:AVERage:COUNt <count>

[:SENSe]:AVERage:COUNt?

Description: Sets/Queries the measurement averaging count. The signal acquisition time is averaging-count multiples of the base value determined by the measurement speed. To execute the averaging, enable the averaging function by [:SENSe]:AVERage[:STATe] command additionally.

Parameter: <count> {numeric value, range 1 to 256}
MAX / MIN can be used.

Setting example: :AVER:COUN 100
Sets the averaging count to 100.

Response: {NR1}

Query example: :AVER:COUN?

Response example: +100
The averaging count is 100.

[:SENSe]:AVERage[:STATe] {ON|OFF|1|0}

[:SENSe]:AVERage[:STATe]?

Description: Sets/Queries whether the averaging is executed or not.
Set the averaging count additionally with [:SENSe]:AVERage:COUNt command.

Parameter: {ON|1} Enables the averaging function.
{OFF|0} Disables the averaging function.

Setting example: :AVER ON
Sets so as to enable the averaging function.

Response: {1|0}

Query example: :AVER?

Response example: 1
The averaging function is enabled.

Remarks: If the averaging count is set to 2 or larger value by the panel operation, the averaging function is automatically enabled. If the averaging count is set to 1 by the panel operation, the averaging function is automatically disabled.

[:SENSe]:CORRection:CKIT:STANdard1:FORMat {GB|CPG}

[:SENSe]:CORRection:CKIT:STANdard1:FORMat?

Description: Sets/Queries the format of OPEN correction value.

Parameter:

GB	G,B (conductance G and susceptance B of parallel equivalent circuit)
CPG	Cp,G (C and G of parallel equivalent circuit)

Setting example: :CORR:CKIT:STAN1:FORM GB

Sets the format of OPEN correction value to G,B.

Response: {GB|CPG}

Query example: :CORR:CKIT:STAN1:FORM?

Response example: GB

The format of OPEN correction value is G,B.

[:SENSe]:CORRection:CKIT:STANdard2:FORMat {RX|LSRS}

[:SENSe]:CORRection:CKIT:STANdard2:FORMat?

Description: Sets/Queries the format of SHORT correction value.

Parameter:

RX	Rs,X (resistance R and reactance X of series equivalent circuit)
LSRS	Ls,Rs (inductance L and R of series equivalent circuit)

Setting example: :CORR:CKIT:STAN2:FORM RX

Sets the format of SHORT correction value to Rs,X.

Response: {RX|LSRS}

Query example: :CORR:CKIT:STAN2:FORM?

Response example: RX

The format of SHORT correction value is Rs,X.

[:SENSe]:CORRection:CKIT:STANdard3[:SPOT] <primary parameter>,<secondary parameter>

[:SENSe]:CORRection:CKIT:STANdard3[:SPOT]?

Description: Sets/Queries the spot LOAD standard value.
Set the format of standard value additionally with [:SENSe]:CORRection:CKIT:STANdard3:FORMat command.

Parameter: <primary parameter>{numeric value, range 0, ± (1E-16 to 9.99999E+11)}
<secondary parameter>{numeric value, range 0, ± (1E-16 to 9.99999E+11)}
Actually set value follows the format of standard value and it is rounded to the range same as set by the panel operation. The unit follows the format of standard value.

Setting example: :CORR:CKIT:STAN3 12.3456E-06,0.01234
Sets the spot LOAD standard value to Cs = 12.3456 µF, D = 0.01234. (when the format is CSD)

Response: <primary parameter>,<secondary parameter> {numeric value, format NR3}
If the standard value is invalid, both parameters are set to 9.9E+37.

Query example: :CORR:CKIT:STAN3?

Response example: +1.23456E-05,+1.23400E-02
Spot LOAD standard value is Cs = 1.23456E-05 F (12.3456 µF), D = 1.23400E-02 (0.01234). (when the format is CSD)

Remarks: Alternatively, you can use the [:SENSe]:CORRection:DATA command to set/query the spot LOAD standard value.

[:SENSe]:CORRection:CKIT:STANdard3:FORMat {CPD|CSD|RCP|RLS|RX|ZPH}

[:SENSe]:CORRection:CKIT:STANdard3:FORMat?

Description: Sets/Queries the format of LOAD standard value and LOAD correction value.

Parameter:

CPD	Cp, D (in either case, the order is primary parameter and secondary parameter)
CSD	Cs,D
RCP	Rp,Cp
RLS	Rs,Ls
RX	Rs,X
ZPH	Z,θ

Setting example: :CORR:CKIT:STAN3:FORM CSD
Sets the format of LOAD standard value and LOAD correction value to Cs,D.

Response: {CPD|CSD|RCP|RLS|RX|ZPH}

Query example: :CORR:CKIT:STAN3:FORM?

Response example: CSD
The format of LOAD standard value and LOAD correction value is Cs,D.

[:SENSe]:CORRection:COLLect[:ACQuire] {STANdard1|STANdard2|STANdard3|STANdard4}

Description: Measures the specified correction value (either OPEN, SHORT, or LOAD). If a correction data is obtained successfully, a setting is made automatically so as to execute that correction.
Or, the LOAD standard value is measured.

Parameter:
STANdard1 OPEN correction value
STANdard2 SHORT correction value
STANdard3 LOAD correction value
STANdard4 LOAD standard value

Setting example: :CORR:COLL STAN2
Measures the SHORT correction value and enable the SHORT correction.

Remarks: To execute the LOAD correction correctly, not only the LOAD correction value, but also the OPEN correction value and SHORT correction value must be measured in advance. Additionally, LOAD standard value must be given. Any measurement order is acceptable.
As this command is an overlap command, if such program is made that *OPC? query is added at the end so as to receive the response message 1, the instrument can wait for the completion of correction.
Example :CORR:COLL STAN2;*OPC?
For details, see the description of the OPC command and OPC? query.
Or, *WAI command may be added to make the instrument wait to execute following commands until the completion of correction.
Example :CORR:COLL STAN2;*WAI
You can also generate SRQ according to the CORR bit of the operation status at the completion of correction.

During the measurement of the correction value in the remote control, the following message is displayed.

```
Warning  Level:2  Code xxxxxx
Correction meas running
```

The first line shows the warning for interrupting the previous operation and does not mean an error of the correction value.
When the correction value is measured normally, the completion message is displayed for a short time.

```
Correction meas completed
```

If the correction value is out of the specified range, the warning message is displayed.

```
Out of range
```

When a correction value was not obtained due to noise, etc., the same warning message is displayed, and the error is recorded on the error queue. Query the error if necessary.

••• ☞ :SYSTem:ERRor?

[:SENSe]:CORRection:COLLect:METHod {REFL2|REFL3}

[:SENSe]:CORRection:COLLect:METHod?

Description: Sets/Queries correction items to be operated by [:SENSe]:CORRection[:STATe] command.

Parameter: REFL2 OPEN correction and SHORT correction

REFL3 OPEN correction, SHORT correction, and LOAD correction

Setting example: :CORR:COLL:METH REFL3

OPEN correction, SHORT correction, and LOAD correction are to be operated.

Response: {REFL2|REFL3}

Query example: :CORR:COLL:METH?

Response example: REFL3

The operation targets are OPEN correction, SHORT correction, and LOAD correction.

Remarks: If REFL2 is set, the LOAD correction is disabled.

[:SENSe]:CORRection:DATA[:SPOT] {STANdard1|STANdard2|STANdard3|STANdard4}, <primary parameter>, <secondary parameter>

[:SENSe]:CORRection:DATA[:SPOT]? {STANdard1|STANdard2|STANdard3|STANdard4}

Description: Sets/Queries the spot OPEN correction value, spot SHORT correction value, spot LOAD correction value, and spot LOAD standard value.

Set the format of the correction value with the following command in advance.

☞ [SENSe]:CORRection:CKIT:STANdard{1|2|3}:FORMat

Parameter: STANdard1 Sets/Queries the spot OPEN correction value.

STANdard2 Sets/Queries the spot SHORT correction value.

STANdard3 Sets/Queries the spot LOAD correction value.

STANdard4 Sets/Queries the spot LOAD standard value.

<primary parameter>{numeric value, range 0, ± (1E-16 to 9.99999E+11)}

<secondary parameter>{numeric value, range 0, ± (1E-16 to 9.99999E+11)}

Actually set value follows the format of correction value and it is rounded to the range same as set by the panel operation. The unit follows the format of correction value.

Setting example: :CORR:DATA STAN2,15.3E-03,2.35E-03

Sets the spot SHORT correction value to Rs = 15.3 mΩ, X = 2.35 mΩ.

(when the format of correction value is RX)

Response: <correction primary parameter>, <correction secondary parameter> {numeric value, format NR3}
If the correction value or standard value is invalid, both parameters are set to 9.9E+37.

Query example: :CORR:DATA? STAN2

Response example: +1.53000E-02,+2.3500E-03
Spot SHORT correction value is Rs = 1.53000E-02 Ω (15.3 mΩ), X = 2.3500E-03 Ω
(2.35 mΩ). (when the format of correction value is RX)

[:SENSe]:CORRection:LIMit:LOWer <lower limit frequency>
[:SENSe]:CORRection:LIMit:LOWer?

Description: Sets/Queries the lower limit frequency when measuring the OPEN / SHORT / LOAD correction values and LOAD standard value globally.

Parameter: <lower limit frequency> {numeric value, range 0.02 to 100E3, resolution 2 digits, unit: Hz}
Initial value = 40 Hz
Suffix K (10^3) and unit Hz can be used.
Example:1KHZ (=1000).

Setting example: :CORR:LIM:LOW 10
Sets the lower limit frequency of correction to 10 Hz.

Response: <lower limit frequency> {numeric value, format NR3}

Query example: :CORR:LIM:LOW?

Response example: +1.00000E+01
The lower limit frequency of correction is 10 Hz.

Remarks: When setting the upper or lower limit, the other is automatically adjusted to meet the relationship of upper limit frequency ≥ lower limit frequency × 2.
Frequency for direct correction value measurement
••• ☞ ZM2376 Instruction Manual (Basics)
"3.5.6.6 Correction of Measurement Error"
Correction values for DC resistance measurement are measured regardless of this setting.

[:SENSe]:CORRection:LIMit:UPPer <upper limit frequency>

[:SENSe]:CORRection:LIMit:UPPer?

Description: Sets/Queries the upper limit frequency when measuring the OPEN / SHORT / LOAD correction values and LOAD standard value globally.

Parameter: <upper limit frequency> {numeric value, range 2E3 to 5.5E6, resolution 2 digits, unit: Hz}

Initial value = 5.5 MHz

Suffix K (10^3) and unit Hz can be used.

Example:1KHZ (=1000).

Setting example: :CORR:LIM:UPP 1E6

Sets the upper limit frequency of correction to 1 MHz.

Response: <upper limit frequency> {numeric value, format NR3}

Query example: :CORR:LIM:UPP?

Response example: +1.00000E+06

The upper limit frequency of correction is 1 MHz.

[:SENSe]:CORRection:LOAD[:STATe] {ON|OFF|1|0}

[:SENSe]:CORRection:LOAD[:STATe]?

Description: Sets/Queries whether LOAD correction is executed or not.

Parameter: {ON|1} Enables the LOAD correction.

{OFF|0} Disables the LOAD correction.

Setting example: :CORR:LOAD ON

Sets so as to enable the LOAD correction.

Response: {1|0}

Query example: :CORR:LOAD?

Response example: 1

The LOAD correction is enabled.

Remarks: When the LOAD correction is enabled, the OPEN correction and SHORT correction are also enabled at the same time.

[:SENSe]:CORRection:OPEN[:STATe] {ON|OFF|1|0}

[:SENSe]:CORRection:OPEN[:STATe]?

Description: Sets/Queries whether OPEN correction is executed or not.

Parameter: {ON|1} Enables the OPEN correction.

{OFF|0} Disables the OPEN correction.

Setting example: :CORR:OPEN ON

Sets so as to enable the OPEN correction.

Response: {1|0}

Query example: :CORR:OPEN?

Response example: 1

The OPEN correction is enabled.

[:SENSe]:CORRection:SHORt[:STATe] {ON|OFF|1|0}
[:SENSe]:CORRection:SHORt[:STATe]?

Description: Sets/Queries whether SHORT correction is executed or not.
Parameter: {ON|1} Enables the SHORT correction.
{OFF|0} Disables the SHORT correction.
Setting example: :CORR:SHOR ON
Sets so as to enable the SHORT correction.
Response: {1|0}
Query example: :CORR:SHOR?
Response example: 1
The SHORT correction is enabled.

[:SENSe]:CORRection:SPOT[:STATe] {ON|OFF|1|0}
[:SENSe]:CORRection:SPOT[:STATe]?

Description: Sets/Queries whether the spot correction or global correction is performed in the OPEN/SHORT/LOAD correction.
Parameter: {ON|1} Enables the spot correction.
This measures the correction value with the current one frequency only.
At the DUT measurement, the spot correction value is used.
{OFF|0} Disables the spot correction.
This measures the correction value in the separately specified frequency range of the global correction.
At the DUT measurement, the global correction value is used.
Setting example: :CORR:SPOT ON
Enables the spot correction.
Response: {1|0}
Query example: :CORR:SPOT?
Response example: 1
The spot correction is enabled.
Remarks: If the measurement frequency is changed when the spot correction is enabled, the OPEN, SHORT, and LOAD corrections are all disabled automatically.

[:SENSe]:CORRection[:STATe] {ON|OFF|1|0}

Description: Sets whether the correction is executed or not to the correction items specified by [:SENSe]:CORRection:COLLect:METHod command.

Parameter: {ON|1} Enables target correction items.
{OFF|0} Disables all the OPEN / SHORT / LOAD corrections.

Setting example: :CORR ON
Enables target correction items.

Remarks: You can also use the [:SENSe]:CORRection:OPEN[:STATe] command to enable/disable each correction.

[:SENSe][:FIMPedance]:APERture[:MODE] {SHORt|MEDium|LONG|RAPid|FAST|SLOW|VSLOw}

[:SENSe][:FIMPedance]:APERture[:MODE]?

Description: Sets/Queries the measurement time or measurement speed.

Parameter: RAPid Sets measurement time to shortest (/ measurement speed to fastest).
{SHORt|FAST} Sets measurement time to short (/ measurement speed to fast).
MEDium Sets measurement time or measurement speed to standard value.
{LONG|SLOW} Sets measurement time to long (/ measurement speed to slow).
VSLOw Sets measurement time to longest (/ measurement speed to lowest).
Specific measurement time••• ☞ ZM2376 Instruction Manual (Basics) "3.5.6.5 Measurement Speed"

Setting example: :APER MED
Sets measurement time (/ measurement speed) to standard.

Response: {RAP|SHOR|MED|LONG|VSLO}

Query example: :APER?

Response example: MED
The measurement time (/ measurement speed) is standard.

[:SENSe][:FIMPedance]:CONTact:VERify {ON|OFF|1|0}

[:SENSe][:FIMPedance]:CONTact:VERify?

Description: Sets/Queries whether the contact check is used or not.

Parameter: {ON|1} Enables the contact check function.

{OFF|0} Disables the contact check function.

Setting example: :CONT:VER ON

Sets so as to enable the contact check.

Response: {1|0}

Query example: :CONT:VER?

Response example: 1

The contact check is enabled.

[:SENSe][:FIMPedance]:CREJect:LIMIt <low capacitance limit>

[:SENSe][:FIMPedance]:CREJect:LIMIt?

Description: Sets/Queries the low capacitance limit used in the low capacitance check.

Parameter: <low capacitance limit> {numeric value, range 0, 0.0001E-12 to 999.999E-9, unit: F}

Setting example: :CREJ:LIMI 5E-12

Sets the low capacitance limit to 5 pF.

Response: <low capacitance limit> {numeric value, format NR3}

Query example: :CREJ:LIMI?

Response example: +5.00000E-12

Low capacitance limit is 5 pF.

[:SENSe][:FIMPedance]:CREJect[:STATe] {ON|OFF|1|0}

[:SENSe][:FIMPedance]:CREJect[:STATe]?

Description: Sets/Queries whether the low capacitance check is used or not.

Parameter: {ON|1} Enables the low capacitance check.

{OFF|0} Disables the low capacitance check.

Setting example: :CREJ ON

Enables the low capacitance check.

Response: {1|0}

Query example: :CREJ?

Response example: 1

Low capacitance check is enabled.

[:SENSe][:FIMPedance]:RANGe:AUTO {ON|OFF|1|0}

[:SENSe][:FIMPedance]:RANGe:AUTO?

Description: Sets/Queries automatic switching of measurement range.

Parameter: {ON|1} Enables automatic switching of measurement range.

{OFF|0} Disables automatic switching of measurement range, and fixes the range (HOLD).

Setting example: :RANG:AUTO ON

Sets so as to enable automatic switching of measurement range.

Response: {1|0}

Query example: :RANG:AUTO?

Response example: 1

Automatic switching of measurement range is enabled.

Remarks: Automatic switching of measurement range is automatically disabled if specific range is set by panel operation or remote control.

[:SENSe][:FIMPedance]:RANGe[:UPPer] <range>

[:SENSe][:FIMPedance]:RANGe[:UPPer]?

Description: Sets/Queries the measurement range. The range is specified in impedance.

Parameter: <range>

{numeric value, range 100E-3|1|10|100|1E+3|10E+3|100E+3|1E+6, unit: Ω}.

As suffix, M (10^{-3}), K (10^{+3}), MEG (10^{+6}) can be used.

As unit, OHM can be used. MAX / MIN can be used.

Parameter example: 10K (=10E+3), 100OHM (=100).

If an out-of-range value is set, it becomes minimum or maximum value.

Range		Measurement range
1 MΩ ≤ Parameter	→	1 MΩ
100 kΩ ≤ Parameter < 1 MΩ	→	100 kΩ
10 kΩ ≤ Parameter < 100 kΩ	→	10 kΩ
1 kΩ ≤ Parameter < 10 kΩ	→	1 kΩ
10 Ω < Parameter < 1 kΩ	→	100 Ω
1 Ω < Parameter ≤ 10 Ω	→	10 Ω
100 mΩ < Parameter ≤ 1 Ω	→	1 Ω
Parameter ≤ 100 mΩ	→	100 mΩ

If an arbitrary value is given, the range suitable for measuring that value is set. Note that the recommended range of 10 Ω and 100 Ω may be different from this table depending on settings. For this reason, do not specify the range, 10 Ω < Parameter < 100 Ω.

Setting example: :RANG 10E+3

Sets the measurement range to 10 kΩ.

Response: {numeric value, format NR3}

Query example: :RANG?

Response example: +1.00000E+04

The measurement range is 1.00000E+04 Ω (10 kΩ).

[:SENSe]:FRESistance:RANGe:AUTO {ON|OFF|1|0}
[:SENSe]:FRESistance:RANGe:AUTO?

Description: Sets/Queries automatic switching of DC resistance measurement range.

Parameter: {ON|1} Enables automatic switching of DC resistance measurement range.

{OFF|0} Disables automatic switching of DC resistance measurement range, and fixes the DC resistance measurement range (HOLD).

Setting example: :FRES:RANG:AUTO ON

Sets so as to enable automatic switching of DC resistance measurement range.

Response: {1|0}

Query example: :FRES:RANG:AUTO?

Response example: 1

Automatic switching of DC resistance measurement range is enabled.

[:SENSe]:FRESistance:RANGe[:UPPer] <range>
[:SENSe]:FRESistance:RANGe[:UPPer]?

Description: Sets/Queries the DC resistance measurement range.

Parameter: <range>

{numeric value, range 100E-3|1|10|100|1E+3|10E+3|100E+3|1E+6, unit: Ω}.

Parameter specifying method is same as that for the measurement range of AC impedance.

For details ☞ [:SENSe][:FIMPedance]:RANGe[:UPPer]

Setting example: :FRES:RANG 100E-3

Sets the measurement range to 100 mΩ.

Response: {numeric value, format NR3}

Query example: :FRES:RANG?

Response example: +1.00000E-01

DC resistance measurement range is 1.00000E-01 Ω (100 mΩ).

Remarks: Automatic switching of DC resistance measurement range is automatically disabled if specific DC resistance measurement range is set. The DC resistance measurement range cannot be specified or fixed from the panel.

[:SENSe]:FUNCtion:CONCurrent {ON|OFF|1|0}

[:SENSe]:FUNCtion:CONCurrent?

Description:	Sets/Queries how many measurement functions, two or only one, are specified. The measurement function are specified with [:SENSe]:FUNCtion[:ON] command.	
Parameter:	{ON|1}	Specifies two measurement functions.
	{OFF|0}	Specifies only one measurement function. (Initial value, *RST)
Setting example:	:FUNC:CONC ON Sets so as to specify two measurement functions.	
Response:	{1|0}	
Query example:	:FUNC:CONC?	
Response example:	1 A setting is made so as to specify two measurement functions.	
Remarks:	You can use the :CALCulate1:FORMat and :CALCulate2:FORMat commands to directly set the primary and secondary parameters without using the [:SENSe]:FUNCtion subsystem. For example, you can set Cp and D directly.	

[:SENSe]:FUNCtion[:ON] <measurement function>

[:SENSe]:FUNCtion[:ON]?

Description: Sets/Queries the measurement functions.
Set in advance the number of measurement functions to be specified with [:SENSe]:FUNCtion:CONCurrent command.

Parameter: 1) When only one measurement function is specified

<measurement function>	<description>
"FIMPedance"	Impedance measurement (series equivalent circuit)
"FADMittance"	Admittance measurement (parallel equivalent circuit)

2) When two measurement functions are specified

<measurement function>	<description>
“FIMPedance","FRESistance"	Impedance measurement (series equivalent circuit) and DC (direct-current) resistance measurement
“FADMittance","FRESistance"	Admittance measurement (parallel equivalent circuit) and DC (direct-current) resistance measurement

Setting example: :FUNC 'FIMP','FRES'
Specifies two measurement functions of impedance measurement and DC resistance measurement.

Response: <measurement function>
{"FIMP"|"FADM"|"FIMP","FRES"|"FADM","FRES"}

Query example: :FUNC?

Response example: "FIMP","FRES"
Two measurement functions of impedance measurement and DC resistance measurement are specified.

Remarks: FIMPedance / FADMittance setting from the remote control and the series equivalent circuit / parallel equivalent circuit setting from the panel are interlocked (same).

:SOURce:CURRent:ALC[:STATe] {ON|OFF|1|0}

:SOURce:CURRent:ALC[:STATe]?

Description: Sets/Queries whether constant current drive is executed or not.

Parameter: {ON|1} Enables the constant current function.
(Control so that monitored value falls within ±1% of set value)
{OFF|0} Disables the constant current function.

Setting example: :SOUR:CURR:ALC OFF
Sets so as to disable the constant current function.

Response: {1|0}

Query example: :SOUR:CURR:ALC?

Response example: 0
The constant current function is disabled.

Remarks: The impedance range capable of driving with constant current is restricted by the measurement range.

Details••• ☞ ZM2376 Instruction Manual (Basics)
"3.5.6.3 Measurement Range"

The constant voltage function is automatically disabled if the constant current function is enabled.
When the constant current function is disabled, the instrument operates following the setting of measurement voltage, not the setting of measurement current.

:SOURce:CURRent[:LEVel][:IMMediate][:AMPLitude] <current>

:SOURce:CURRent[:LEVel][:IMMediate][:AMPLitude]?

Description: Sets/Queries the measurement current level when constant current drive is executed.

Parameter: <current> {numeric value, range 1E-6 to 200E-3, resolution 3 digits (0.1E-6 for numeric value < 10E-6), unit: Arms}
Suffix U (10^{-6}) and M (10^{-3}), and unit A can be used.
Example:100U (=1E-4), 10MA (=0.01)

Setting example: :SOUR:CURR 2E-3
Sets the measurement current level to 2 mArms.

Response: <current> {numeric value, format NR3}

Query example: :SOUR:CURR?

Response example: +2.00000E-03
The measurement current level is 2.00000E-03 Arms (2 mArms).

:SOURce:FREQuency[:CW] <frequency>

:SOURce:FREQuency[:CW]?

Description: Sets/Queries the measurement frequency.

Parameter: <frequency>{numeric value, range 20E-3 to 5.00000E6, resolution 6 digits (<1 mHz when 100 Hz), unit: Hz}
Suffix K (10^3), unit Hz, and MAX / MIN can be used.
Example:0.12K (=120), 1KHZ (=1E3).

Setting example: :SOUR:FREQ 1000
Sets the measurement frequency to 1 kHz.

Response: <frequency> {numeric value, format NR3}

Query example: :SOUR:FREQ?

Response example: +1.00000E+03
The measurement frequency is 1.00000E+03 Hz (1 kHz).

:SOURce:RESistance[:LIMit]:LOWer <resistance>

:SOURce:RESistance[:LIMit]:LOWer?

Description: Sets/Queries the minimum output impedance of drive signal source.

Parameter: <resistance> {numeric value, range 6, 25, 100, unit: Ω}
As unit, OHM can be used. Parameter example: 6OHM (=6).
If an arbitrary value is given, it becomes a nearest value.

Setting example: :SOUR:RES:LOW 25
Set the minimum output impedance to 25 Ω.

Response: <resistance> {numeric value, format NR3}

Query example: :SOUR:RES:LOW?

Response example: +2.50000E+01
The minimum output impedance is 25 Ω.

Remarks: When selecting 25 Ω, use the :SOURce:RESistance[:LIMit]:ZRANge command to specify the upper limit of the applied measurement range.

:SOURce:RESistance[:LIMit]:ZRANge <range>

:SOURce:RESistance[:LIMit]:ZRANge?

Description: Sets/Queries the maximum measurement range to which the output impedance 25 Ω is applied.

Parameter: <range> {numeric value, range 1 or 100, unit: Ω}
As unit, OHM can be used. Parameter example: 100OHM (=100).
If an arbitrary value is given, it becomes a nearest value.

Setting example: :SOUR:RES:ZRAN 100
Sets the maximum measurement range to which the output impedance 25 Ω is applied to 100 Ω.

Response: <resistance> {numeric value, format NR3}

Query example: :SOUR:RES:ZRAN?

Response example: +1.00000E+02

The maximum measurement range to which the output impedance 25 Ω is applied is 100 Ω.

:SOURce:VOLTage:ALC[:STATe] {ON|OFF|1|0}

:SOURce:VOLTage:ALC[:STATe]?

Description: Sets/Queries whether constant voltage drive is executed or not.

Parameter: {ON|1} Enables the constant voltage function.
(Control so that monitored value falls within ±1% of set value)
{OFF|0} Disables the constant voltage function.

Setting example: :SOUR:VOLT:ALC OFF
Sets so as to disable the constant voltage function.

Response: {1|0}

Query example: :SOUR:VOLT:ALC?

Response example: 0
The constant voltage function is disabled.

Remarks: The range in which ALC (automatic level control) functions actually is limited.

Details••• ☞ ZM2376 Instruction Manual (Basics)
"3.5.6.3 Measurement Range" and
"4.1 Measuring with Specific Voltage or Current (ALC)"

The constant current function is automatically disabled if the constant voltage function is enabled.

:SOURce:VOLTage[:LEVel][:IMMediate][:AMPLitude] <voltage>

:SOURce:VOLTage[:LEVel][:IMMediate][:AMPLitude]?

Description: Sets/Queries the measurement voltage level.

Parameter: <voltage> {numeric value, range 0.010 to 5.00, setting resolution 3 digits (0.001 when <1), unit: Vrms}
Suffix M (10^{-3}), and unit V , MAX/MIN can be used.
Example:1000M (=1), 500MV (=0.5)

Setting example: :SOUR:VOLT 0.5
Sets the measurement voltage level to 0.5 Vrms.

Response: <voltage> {NR3}

Query example: :SOUR:VOLT?

Response example: +5.00000E-01
The measurement voltage level is 5.00000E-01 Vrms (0.5 Vrms).

:SOURce:VOLTage[:LEVel][:IMMediate]:OFFSet <DC offset>
:SOURce:VOLTage[:LEVel][:IMMediate]:OFFSet?

Description: Sets/Queries internal DC bias voltage (DC offset) superimposed on the measurement signal. To actually superimpose the DC offset, set additionally the following command so as to superimpose.
••• ☞ :SOURce:VOLTage[:LEVel][:IMMediate]:OFFSet:STATe

Parameter: <DC offset> {numeric value, range 0 to 5.000, resolution 0.001, unit: V}
Unit V and MAX / MIN can be used. Example:0.5V (=0.5).

Setting example: :SOUR:VOLT:OFFS 1.5
Sets the internal DC bias voltage to 1.5 V.

Response: <DC offset> {numeric value, format NR3}

Query example: :SOUR:VOLT:OFFS?

Response example: +1.50000E+00
Internal DC bias voltage is +1.50000E+00 V (1.5 V).

:SOURce:VOLTage[:LEVel][:IMMediate]:OFFSet:AUTO {ON|OFF|1|0|TRACking}
:SOURce:VOLTage[:LEVel][:IMMediate]:OFFSet:AUTO?

Description: Sets/Queries whether auto DC balance function and auto DC cancel function are used or not. When internal DC bias voltage is not superimposed, auto DC balance function and auto DC cancel function do not work.
••• ☞ :SOURce:VOLTage[:LEVel][:IMMediate]:OFFSet:STATe

Parameter:
{OFF|0} Disables auto DC balance function and auto DC cancel function.
{ON|1} Enables auto DC balance function and auto DC cancel function.
TRACking Enables only auto DC cancel function.

Setting example: :SOUR:VOLT:OFFS:AUTO 1
Enables the auto DC balance and auto DC cancel functions.

Response: {1|0|TRAC}

Query example: :SOUR:VOLT:OFFS:AUTO?

Response example: 1
Auto DC bias function and auto DC cancel function are enabled.

:SOURce:VOLTage[:LEVel][:IMMediate]:OFFSet:STATe {ON|OFF|1|0}

:SOURce:VOLTage[:LEVel][:IMMediate]:OFFSet:STATe?

Description: Sets whether internal DC bias voltage (DC offset) is superimposed on the measurement signal.

Parameter: {ON|1} Internal DC bias voltage is superimposed.

{OFF|0} Internal DC bias voltage is not superimposed.

Setting example: :SOUR:VOLT:OFFS:STAT 1

Sets so as to superimpose the internal DC bias voltage.

Response: {1|0}

Query example: :SOUR:VOLT:OFFS:STAT?

Response example: 1

Superimpose internal DC bias voltage.

:SOURce:VOLTage:MODE {CONTinuous|SYNChronous}

:SOURce:VOLTage:MODE?

Description: Sets/Queries the triggered drive.

Parameter: CONTinuous Triggered drive is disabled and measurement signal is output continuously.

SYNChronous Triggered drive is enabled and measurement signal is output only during measurement.

Setting example: :SOUR:VOLT:MODE SYNC

Sets so as to output the measurement signal only during measurement in synchronization with the trigger.

Response: {CONT|SYNC}

Query example: :SOUR:VOLT:MODE?

Response example: SYNC

The measurement signal is output only during measurement.

:STATus:OPERation:CONDition?

Description: Queries the operation condition register (OPCR).
Response: {numeric value, format NR1, range 0 to 32767}
Query example: :STAT:OPER:COND?
Response example: +0
The content of operation condition register is 0.

:STATus:OPERation:ENABle <mask>

:STATus:OPERation:ENABle?

Description: Sets/queries the operation event enable register (OPEE).
Parameter: <mask> {numeric value, range 0 to 65535}
Out of range value causes an error.
Setting example: :STAT:OPER:ENAB 0
Sets so as to disable the occurrence of operation event.
Response: {numeric value, format NR1, range 0 to 32767}
Query example: :STAT:OPER:ENAB?
Response example: +0
The content of operation event enable register is 0 (all disabled).

:STATus:OPERation[:EVENt]?

Description: Queries the operation event register (OPER).
Response: {numeric value, format NR1, range 0 to 32767}
Query example: :STAT:OPER?
Response example: +0
The content of operation event register is 0 (no event).

For details of operation status, see “6.4 Operation Status”.

:SYSTem:ADELay <delay time>

:SYSTem:ADELay?

Description: Sets/Queries the trigger delay time in individual measurement during automatic adjustment.
This parameter is used in the ALC, automatic range selection, automatic bias adjustment, etc.

Parameter: <delay time> {numeric value, range 1E-3 to 99E-3, unit: s, resolution 1 ms}
Initial value = 20 ms

Setting example: :SYST:ADEL 10E-3
Sets the trigger delay time during automatic adjustment to 10 ms.

Response: <delay time> {numeric value, format NR3, unit: s}

Query example: :SYST:ADEL?

Response example: +1.00000E-02
Trigger delay time during automatic adjustment is 10 ms.

:SYSTem:ERRor?

Description: Queries the error content.

Response: <error number>, <error message>
<error number> {numeric value, format NR1, range -32768 to +32767}
<error message> {format SRD}
Details ☞ "9. ERROR MESSAGES"

Query example: :SYST:ERR?

Response example: +0,"No error"
No error is found.

Remarks: The oldest error recorded in the error queue is read.
The content of read error is cleared from the error queue.
Maximum 16 errors can be saved in the error queue. If exceeding this capacity, the 16th error message changes to "Queue overflow", and error messages after that are dumped.

:SYSTem:KLOCk {ON|OFF|1|0}

:SYSTem:KLOCk?

Description: Sets/Queries the key lock function of the front panel.

Parameter: {ON|1} Turns on the key lock function (key operations are disabled).
{OFF|0} Turns off the key lock function (key operations are enabled).

Setting example: :SYST:KLOC 1
Sets so as to turn on the key lock (key operations are disabled).

Response: {1|0}

Query example: :SYST:KLOC?

Response example: 1
The key lock is turned on (key operations are disabled).

:SYSTem:MEMory {SPOT|PART|SETTing|CORRection|BOTH}

:SYSTem:MEMory?

Description: Sets/Queries the target to be recalled from the memory through selection line of handler interface.

Parameter:

SPOT	Spot correction values (OPEN / SHORT / LOAD)
PART	Main settings and the spot correction value
SETTing	All settings (measurement conditions such as frequency)
CORRection	All correction values (global and spot)
BOTH	(Both all settings and all correction values)

Setting example: :SYST:MEM CORR

Sets correction values as the target to be recalled.

Response: {SPOT|PART|SETT|CORR|BOTH}

Query example: :SYST:MEM?

Response example: CORR

Recall targets are all correction values.

:SYSTem:RCL <memory number>,{SPOT|PART|SETTing|CORRection|BOTH}

Description: Recalls the settings or correction values from the specified memory.

Parameter: <memory number>{numeric value, range 0 to 31}

Out of range value causes an error.

Recall targets are as follows:

SPOT	Spot correction values (OPEN / SHORT / LOAD)
PART	Main settings and the spot correction value
SETTing	Settings (measurement conditions such as frequency)
CORRection	Correction values (OPEN, SHORT, LOAD)
BOTH	(Both setting and correction value)

Setting example: :SYST:RCL 2,BOTH

Recalls both settings and correction values from the memory number 2.

When only the correction values are recalled, the correction is not turned on automatically.

Remarks: The panel does not support operations equivalent to SPOT and PART.

:SYSTem:RST

Description: Initializes the current setting, current correction value, setting/correction value memory, and multi-measurement list.

This operation cannot be performed on the panel. The trigger polarity of the handler interface and the remote control interface related settings are not initialized.

Setting example: :SYST:RST

:SYSTem:SAVE <memory number>,{SETTing|CORRection|BOTH}

Description: Saves the settings and correction values in the specified setting/correction value memory.

Parameter: <memory number>{numeric value, range 0 to 31}
Out of range value causes an error.
SETTing Saves the settings (measurement conditions such as frequency).
CORRection Saves the correction values (OPEN, SHORT, LOAD).
BOTH Saves both settings and correction values.

Setting example: :SYST:SAVE 2,CORR
Saves the correction values in the memory number 2.

------ :TEST:HANDler subsystem command ------

You can use commands and queries in this group to test signals of the handler interface. You can also use the handler interface as the general-purpose digital I/O port. To output signals with the :TEST:HANDler subsystem command, it must be allowed by the :TEST:HANDler:MODE command.

:TEST:HANDler:BIN <bin specification>

Description: Sets the bin sorting signal output of the handler interface.

Parameter: <bin specification> {numeric value, range 0 - 16}

Only the specified one signal is made 1 (low level).

Bin extension = disabled		Bin extension = enabled	
Bin specification	Signal (negative logic)	Bin specification	Signal (negative logic)
0	/OUT OF BINS	0	/OUT OF BINS
1 to 9	/BIN1 to /BIN9	1 to 14	/BIN1 to /BIN14
10	/S-NG	15	/S-NG
11	All of the above signals are 0 (High level)	16	All of the above signals are 0 (High level)

Bin extension ☞ :CALCulate:COMParator:EXTension[:STATe] command

Setting example: :TEST:HAND:BIN 2

Sets the /BIN2 signal to 1 (low level).

:TEST:HANDler:COMP {PHI|PLO|SNG|OFF}

Description: Sets the comparison signal output of the handler interface.

Parameter:
- PHI Sets the /PHI signal to 1 (low level).
- PLO Sets the /PLO signal to 1 (low level).
- SNG Sets the /S-NG signal to 1 (low level).
- OFF Sets all of the /PHI, /PLO, and /S-NG signals to 0 (high level).

Unspecified two signals among three are set to 0 (high level).

Setting example: :TEST:HAND:COMP PHI

Sets the /PHI signal to 1 (low level).

Remarks: When the bin extension is enabled, overwrite the /BIN13 and /BIN14 signals which share a terminal. Oppositely, the output by this command is overwritten by the :TEST:HANDler:BIN command.

:TEST:HANDler:KEYLock?

Description: Queries the /KEY_LOCK input status of the handler interface.

Response: {1|0}

1: low level, 0: high level (negative logic).

Query example: :TEST:HAND:KEYL?

Response example: 1

/KEY_LOCK signal is 1 (low level).

:TEST:HANDler:MODE {ON|OFF|1|0|ERR}

Description: Sets the test mode of the handler interface.

Parameter:

{ON|1}	Enables the test mode. Output signal of the handler interface is controlled by the :TEST:HANDler subsystem command. The comparator result or control signal of the normal comparator is not output. ZM2376 ignores input signals of the handler interface and operates as if nothing were connected to the handler interface.
{OFF|0}	Disables the test mode (initial value). Output by the :TEST:HANDler subsystem command is disabled. I/O of the handler interface is returned to the normal operation. Even if the test mode is disabled, query of input signal is allowed.
ERR	Changes only /ERR signal to the test mode. /ERR signal is controlled by the :TEST:HANDler:STATus:ERR command and the normal error status is not output. Other I/O signals change to the normal operation.

Setting example: :TEST:HAND:MODE 1

Enables the test mode.

:TEST:HANDler:RCL?

Description: Queries the setting/correction value memory selection input status of the handler interface.

Response: {numeric value, range 0 to 127}

7 bits (negative logic) from /RCL6 (MSB) to /RCL0 (LSB) are returned in decimal notation.

Query example: :TEST:HANDler:RCL?

Response example: +2

Setting/Correction value memory number is 2.

Signal from /RCL6 to /RCL0 is 0000010 in binary notation, and the signal level is HHHHHLH (H: high level, L: low level).

:TEST:HANDler:RCLValid?

Description: Queries the /RCL-VALID input status of the handler interface.

Response: {1|0}

1: low level, 0: high level (negative logic).

Query example: :TEST:HAND:RCLV?

Response example: 0

/RCL-VALID signal is 0 (high level).

:TEST:HANDler:STATus:EOM {ON|OFF|1|0}

Description: Sets the /EOM output of the handler interface.
Parameter: {ON|1} Sets the /EOM signal to 1 (low level).
{OFF|0} Sets the /EOM signal to 0 (high level).
Setting example: :TEST:HAND:STAT:EOM ON
Sets the /EOM signal to 1 (low level).

:TEST:HANDler:STATus:INDex {ON|OFF|1|0}

Description: Sets the /INDEX output of the handler interface.
Parameter: {ON|1} Sets the /INDEX signal to 1 (low level).
{OFF|0} Sets the /INDEX signal to 0 (high level).
Setting example: :TEST:HAND:STAT:IND OFF
Sets the /INDEX signal to 0 (high level).

:TEST:HANDler:STATus:NC {ON|OFF|1|0}

Description: Sets the /NC output of the handler interface.
Parameter: {ON|1} Sets the /NC signal to 1 (low level).
{OFF|0} Sets the /NC signal to 0 (high level).
Setting example: :TEST:HAND:STAT:NC 1
Sets the /NC signal to 1 (low level).
Remarks: When the bin extension is enabled, overwrite the /BIN12 signal which shares a terminal. Oppositely, the output by this command is overwritten by the :TEST:HANDler:BIN command.

:TEST:HANDler:STATus:ERR {ON|OFF|1|0}

Description: Sets the /ERR output of the handler interface.
Parameter: {ON|1} Sets the /ERR signal to 1 (low level).
{OFF|0} Sets the /ERR signal to 0 (high level).
Setting example: :TEST:HAND:STAT:ERR 0
Sets the /ERR signal to 0 (high level).

:TEST:HANDler:TRIGger?

Description: Queries the TRIG (/TRIG) input status of the handler interface.
Response: {1|0}
1: high level for TRIG (positive logic), low level for /TRIG (negative logic)
0: low level for TRIG (positive logic), high level for /TRIG (negative logic)
Query example: :TEST:HAND:TRIG?
Response example: 0
Low level for TRIG (positive logic), High level for /TRIG (negative logic).

:TRIGger:DELay <delay time>

:TRIGger:DELay?

Description:	Sets/Queries the trigger delay time.
	Trigger delay time: Time from the trigger to the start of measurement.
Parameter:	{numeric value, range 0.0000 - 999.9999, resolution 0.0001, unit: s}
	Suffix M (10^{-3}), unit S, MAX / MIN can be used.
	Example:10M (=0.010), 200MS (=0.200).
Setting example:	:TRIG:DEL 0.02
	Sets the trigger delay time to 20 ms.
Response:	{numeric value, format NR3, resolution 7 digits}
Query example:	:TRIG:DEL?
Response example:	+2.000000E-02
	The trigger delay time is 2.000000E-02 s (20 ms).

:TRIGger[:IMMediate]

Description:	Applies the trigger to perform measurement once, when the instrument waits for trigger.
	An error will occur when the trigger source is INT or the instrument does not wait for trigger.
Setting example:	:TRIG
	Apply the trigger.

:TRIGger:SOURce {INTernal|MANual|EXTernal|BUS}

:TRIGger:SOURce?

Description:	Sets/queries the trigger source.	
Parameter:	INTernal	Internal trigger (trigger is continuously applied automatically)
	MANual	TRIG key on the front panel
	EXTernal	TRIG signal through handler interface
	BUS	*TRG command, GET message of remote control
Setting example:	:TRIG:SOUR EXT	
	Sets EXT as a trigger source.	
Response:	{INT|MAN|EXT|BUS}	
Query example:	:TRIG:SOUR?	
Response example:	EXT	
	The trigger source is EXT.	

6. STATUS SYSTEM

6.1 Outline of Status System

The status system for ZM2376 is shown in Figure 6-1.

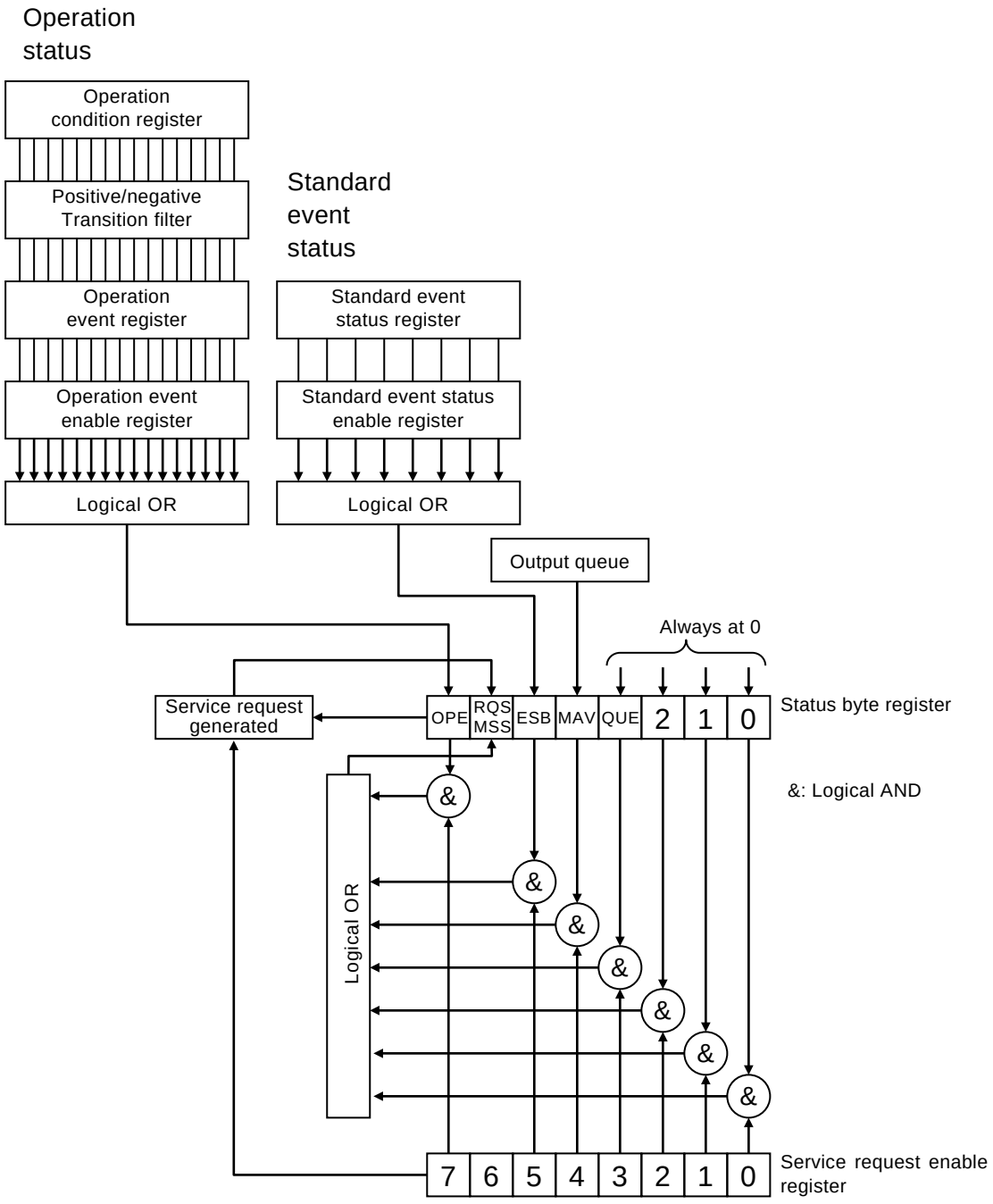


Figure 6-1 Status system

6.2 Status Byte

The definition of the status byte register is shown in Table 6-1. The status byte can be read by serial polling or *STB? query. However, bit 6 becomes RQS (Request Service).

Table 6-1 Status byte register definition

Bit		Weight	Condition for setting to 1	Condition for resetting to 0
OPE	7	128	When any valid bit of the operation status event register is set to 1.	・ When device clear was received ・ After status byte was read
RQS	6	64	When SRQ is sent	・ When device clear was received ・ When status byte was read by serial polling
MSS				・ When device clear was received ・ When master summary bits were all cleared to 0
ESB	5	32	When any valid bit of the standard event status register is set to 1.	When all the valid bits of the standard event status register are set to 0.
MAV	4	16	When the response to the query is ready to be output.	When all of the responses have been output and there remains not to be output.
QUE	3	8	-	Always at 0 (unused)
-	2	4	-	Always at 0 (unused)
-	1	2	-	Always at 0 (unused)
-	0	1	-	Always at 0 (unused)

■ Related commands / queries

*STB?

Queries the content of status byte register.

Bit 6 is MSS (Master Summary Status).

*SRE / *SRE?

Sets/Queries the service request enable register.

Cleared to 0 upon powering on. To clear this register to 0, set it to 0. When bits of the service request enable register are set to 1, the corresponding bits of the status byte register become valid. If at least one valid bit is set to 1, the service request is generated.

The parameter value of the setting message or response message to each register is the sum of the weight of all the bits having 1 as the value.

■ About verifying status when querying

Normally, once a query command is transmitted, you can receive the response correctly, if response message is received. There is no need to check the MAV bit of the status byte. When a processing is progressed while checking the MAV bit, after the query was transmitted, check by serial polling that the MAV bit of status byte becomes 1, and then read the response message, and after checking that the MAV bit becomes 0, perform the next operation.

6.3 Standard Event Status

The standard event status structure is shown on Figure 6-2. Details about status are shown in Table 6-2. When bits of the standard event status enable register are set to 1, the corresponding bits of the standard event status register become valid. When at least one valid bit is set to 1, the status bit register ESB bit is set to 1.

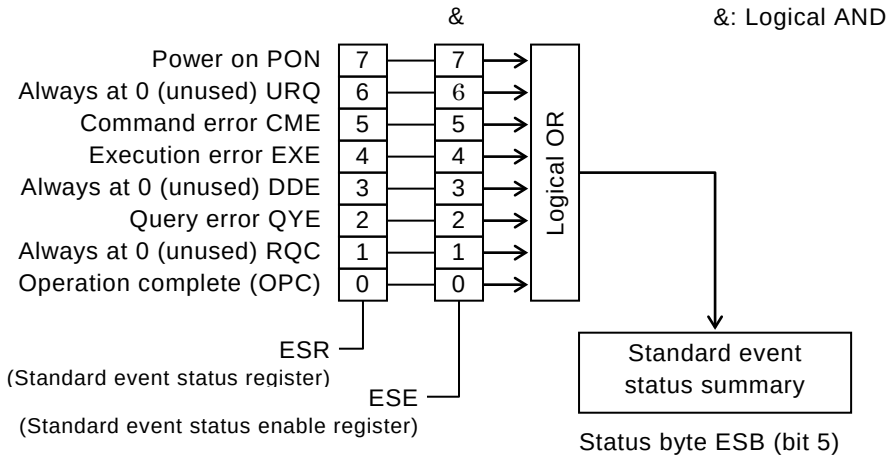


Figure 6-2 Standard event status structure

Table 6-2 Contents of standard event status register

Bit		Weight	Content
PON	7	128	Power on Set to 1 when the power is on. If cleared to 0 by a register readout, remains at 0 until the power is turned on again.
URQ	6	64	User request Always at 0 (unused).
CME	5	32	Command error Set to 1 when there is a syntax error in program code.
EXE	4	16	Execution error Set to 1 when parameters are set beyond possible range values or when settings are contradictory.
DDE	3	8	Device dependent error Set to 1 when the error queue overflows.
QYE	2	4	Query error Set to 1 when a readout has been attempted on an empty response message output buffer or when the response message output buffer data has been lost.
RQC	1	2	Request control Always at 0 (unused).
OPC	0	1	Operation complete Set to 1 when a processing of all commands up to *OPC command completed.

■ Related commands / queries

*ESR?

Queries the standard event status register contents.

Cleared to 0 upon query. Can be cleared also by *CLS command.

Cleared to 0 upon powering on. However the PON bit is set to 1.

*ESE / *ESE?

Sets/queries the standard event status enable register.

Set to 0 to clear the enable register to 0.

Cannot be cleared by any other command.

Cleared to 0 upon powering on.

The parameter value of the setting message or response message to each register is the sum of the weight of all the bits having 1 as the value.

6.4 Operation Status

The operation status structure is shown on Figure 6-3.

As can be seen on Table 6-3, the operation condition register indicates ZM2376 state. The transition filter detects a condition change and generates an event. The filter setting of ZM2376 is fixed. The operation event register retains the events that occurred. When bits of the operation event enable register are set to 1, the corresponding bits of the operation event register become valid. When at least one valid bit is set to 1, the status byte OPE bit is set to 1.

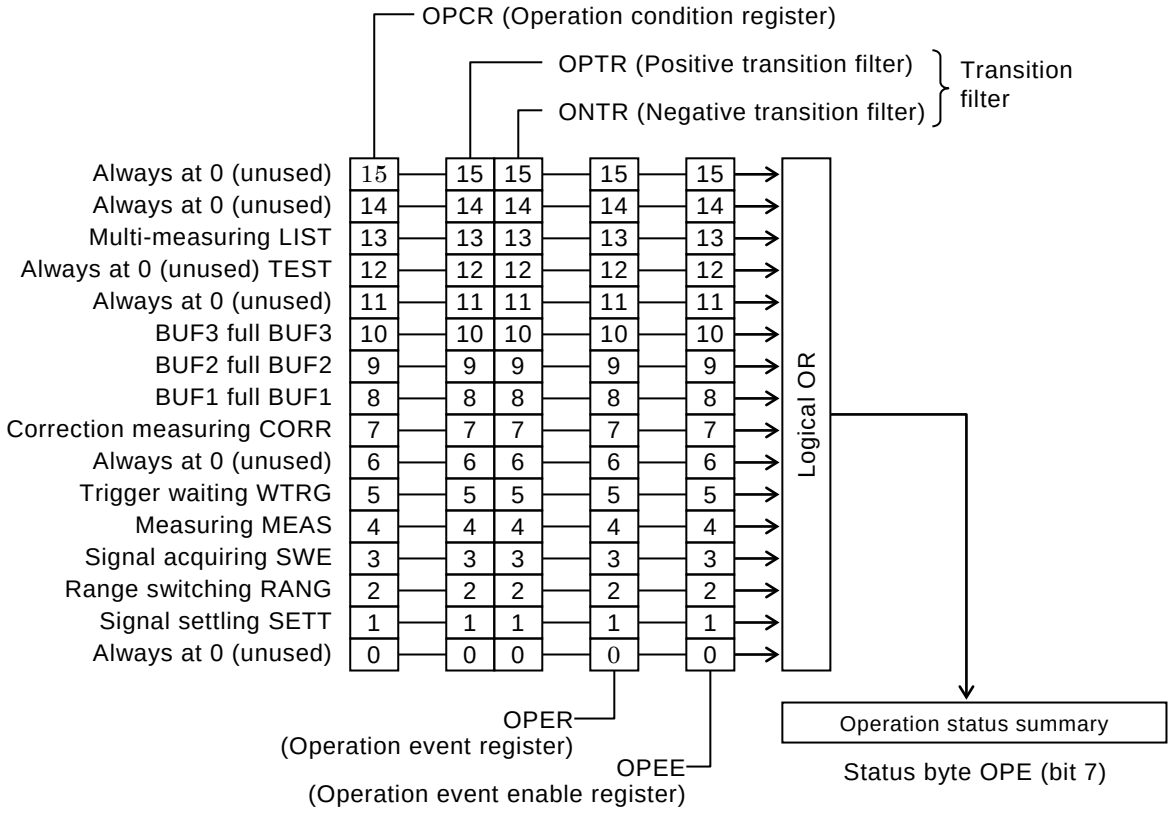


Figure 6-3 Operation status structure

The bit 13 (LIST) of the condition register keeps 1 from the start of multi-measurement to the completion of the measurement in all valid steps.

Table 6-3 Contents of operation condition register and event register

Bit		Weight	Content of condition register (1 when specified status is fulfilled)	Content of event register (Condition for setting to 1)
- -	15 14	32768 16384	Always 0 (unused)	Always 0
LIST	13	8192	Multi-measuring	Multi-measurement completed
TEST	12	4096	Always 0 (unused)	Always 0
-	11	2048	Always 0 (unused)	Always 0
BUF3	10	1024	BUF3 is full	BUF3 becomes full
BUF2	9	512	BUF2 is full	BUF2 becomes full
BUF1	8	256	BUF1 is full	BUF1 becomes full
CORR	7	128	OPEN / SHORT / LOAD correction is being measured	Correction measurement completed
WARM	6	64	Always 0 (unused)	Always 0
WTRG	5	32	Trigger waiting	Device waits for trigger
MEAS	4	16	Measuring (/EOM signal of handler interface is on high level)	Measurement completed
SWE	3	8	Signal acquiring (/INDEX signal of handler interfaced is on high level)	Signal acquisition completed
RANG	2	4	Range switching by automatic range selection function	Measurement range has been established
SETT	1	2	Signal settling (trigger is being delayed)	Signal has settled (trigger delay finished)
-	0	1	Always 0 (unused)	Always 0

■ Related commands / queries

:STATus:OPERation:CONDition?

Queries the operation condition register contents.

The condition register contents are not cleared to 0 even if queried.

The instrument state is always indicated.

:STATus:OPERation[:EVENt]?

Queries operation event register.

The event register is cleared to 0 if queried.

The event register can also be cleared with a *CLS command.

Cleared to 0 upon powering on.

:STATus:OPERation:ENABle / STATus:OPERation:ENABle?

Sets/Queries the operation event enable register.

Set to 0 to clear the enable register to 0.

Cannot be cleared by any other command.

Cleared to 0 upon powering on.

The parameter value of the setting message or response message to each register is the sum of the weight of all the bits having 1 as the value.

7. TRIGGER SYSTEM

The trigger system of ZM2376 is shown below.

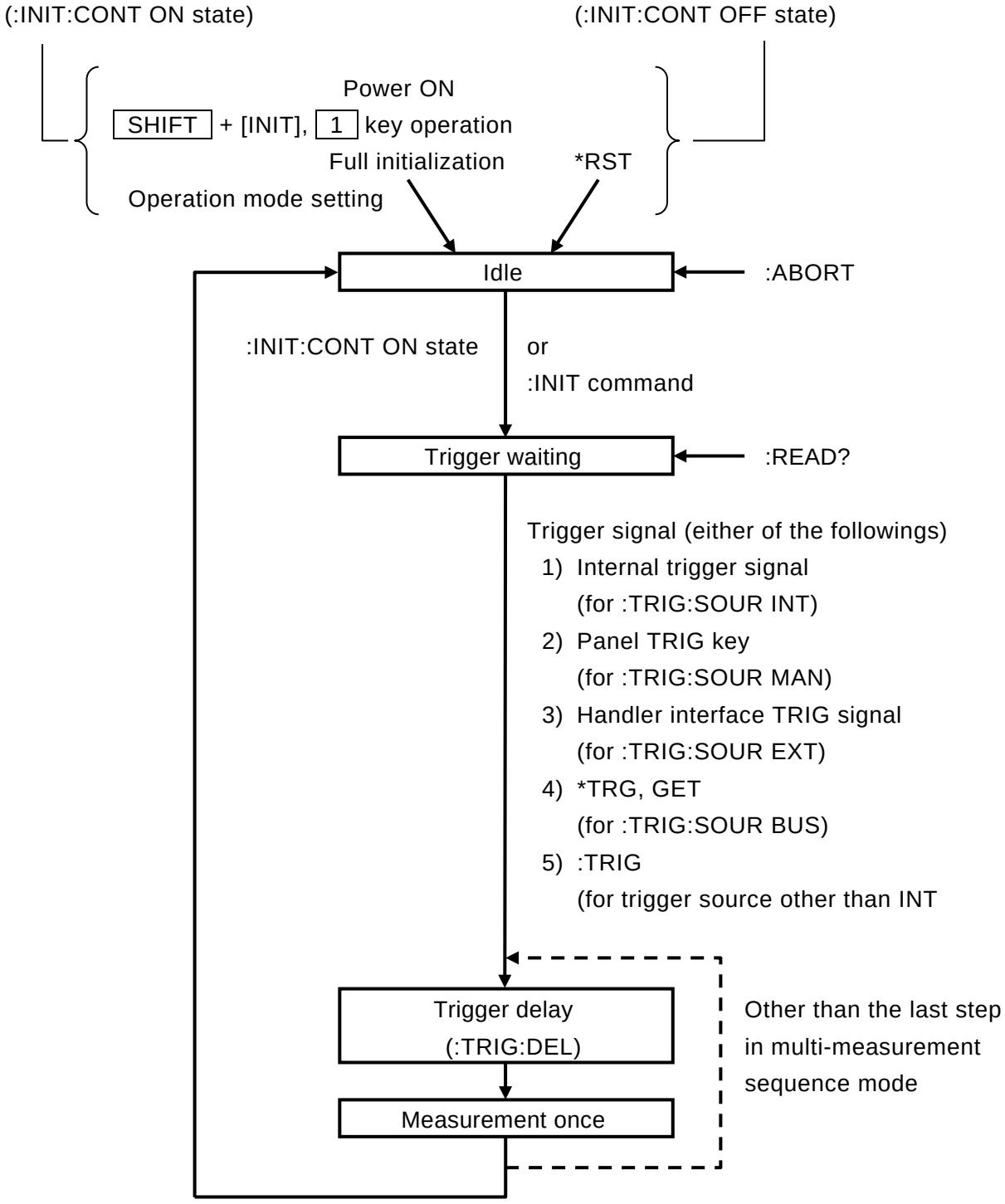


Figure 7-1 Trigger system

Here are typical examples in which a trigger is applied to ZM2376 to execute measurement once to obtain the measured result, and this operation is repeated.

Example 1 Measured value is obtained by *TRG command

```
Power ON                          ' Setting equivalent to :INIT:CONT ON.
Transmit (":INIT:CONT ON")        ' Specify definitely when it is not known exactly how
                                    after start. (Note 1)
Transmit (":TRIG:SOUR BUS")       ' Set to apply the trigger by remote control.
Transmit (":ABORT")               ' → Idle → Trigger waiting state.
Transmit ("*TRG")                 ' Apply trigger, and transfer measured result.
Receive (measurement status, primary parameter measured value, secondary parameter measured
value)
```

Note 1: If :INIT:CONT OFF, after the first measurement, the instrument remains in idle state, and thus the trigger at the second and subsequent times becomes ineffective.

Example 2 Measured value is obtained by :TRIG command and :FETCH? query

```
Transmit ("*RST")                 ' Setting equivalent to :INIT:CONT OFF.
Transmit (":TRIG:SOUR EXT")       ' Set trigger source to handler interface.
Transmit (":INIT")                ' Device goes in trigger waiting state.
Transmit (":TRIG")                ' Trigger can also be applied from handler interface,
                                    omitting this command.
Transmit (":FETC?")               ' Query the latest measured value.
                                    When the trigger is applied and one-time measurement
                                    finished, the result is transmitted.
Receive (measurement status, primary parameter measured value, secondary parameter measured
value)
```

For the limit comparison, :CALC1:LIM:FAIL? can also be used instead of :FETC?

Example 3 Measured value is obtained by READ? query

```
Transmit ("*RST")                 ' Setting equivalent to :INIT:CONT OFF.
Transmit (":TRIG:SOUR EXT")       ' Set trigger source to handler interface.
Transmit (":INIT:CONT ON")        ' Set so as to return to trigger waiting after measurement
                                    completed.
Transmit (":READ?")               ' → Idle → In trigger waiting state,
                                    wait for trigger. (Note 2)
                                    When trigger is applied and one-time measurement
                                    finished, the result is transmitted.
Receive (measurement status, primary parameter measured value, secondary parameter measured
value)
```

Note 2: If :READ? query is executed, the instrument does not execute the next command until the measured result is stored completely in the transmission buffer. The :TRIG command of a program message ":READ?;:TRIG" is executed after READ? query started and the measurement started by TRIG signal of handler interface finished.

8. IMPORTING DATA USING MEASURED DATA BUFFER

Instead of "importing measured data for each measurement", you can also "import measured data at once after saving two or more measurement data in the measured data buffer".
Here is a typical example of measurement using the measured data buffer.

```
Transmit (":TRIG:SOUR EXT")
        ' Set the outside trigger source.
Transmit (":DATA:POIN   BUF1, 100")
        ' Specify the capacity of measured data buffer to be used.
        ' The measured data buffer is cleared now.
Transmit (":DATA:FEED   BUF1,""CALC1"" ")
        ' Specify the measured data buffer to be used and data to be transmitted.
        ' Handling of "" in the transmit command string
        ' depends on the language specification of the controller.
Transmit (":DATA:FEED:CONT   BUF1, ALW")
        ' Set the measured data to be transmitted to the measured data buffer.
        ' Transmit (":INIT") ' Move to trigger waiting state, if necessary.
Transmit (":TRIG")
        ' Apply the trigger to measure.
        ' You can also use the trigger signal of the handler interface or the manual trigger.
        <Monitoring completion of measurement>
        ' Detect the completion by counting times, monitoring the buffer full, etc.
        ' You can also generate SRQ according to the operation status.
        Transmit (":DATA?   BUF1")
        ' Request data from the measured data buffer.
Receive (measured data 1, measured data 2, .. , measured data n)
```

9. ERROR MESSAGES

This section describes main errors that occur in the remote control.

Table 9-1 Error messages

Error No.	Error Message	Description of error
0	No error	------
-100	Command error	Command is not correct. (No detailed classification)
-102	Syntax error	Unrecognizable command or data was received.
-104	Data type error	The format of parameter is improper.
-108	Parameter not allowed	Too many parameters are used, or a parameter is used in illegal position.
-109	Missing parameter	The number of parameters is deficient.
-110	Command header error	Command header is not correct. (No detailed classification)
-113	Undefined header	Undefined command header is used.
-120	Numeric data error	Numeric data is not correct. (No detailed classification)
-130	Suffix error	Suffix (multiplier, unit) is not correct. (No detailed classification)
-140	Character data error	Character data is not correct. (No detailed classification)
-144	Character data too long	Character data is too long (> 12 characters).
-150	String data error	Character string data is not correct. (No detailed classification)
-200	Execution error	Command cannot be executed. (No detailed classification)
-211	Trigger ignored	Trigger was received but could not be executed.
-221	Settings conflict	Command could not be executed due to the restriction between plural settings.
-222	Data out of range	Data is out of allowable range.
-231	Data questionable	An error occurred during measurement, and OPEN / SHORT / LOAD correction values were not obtained.
-241	Hardware missing	Command cannot be executed because hardware to be operated is not installed.
-300	Device-specific error	Command cannot be executed because of instrument-specific internal error. (No detailed classification)
-310	System error	Device-specific internal error occurred. (For instance, loss of memory data)
-330	Self-test failed	An error was found in the self-diagnosis.
-350	Queue overflow	The error queue overflows, and it cannot retain new error.
-363	Input buffer overrun	The input buffer overflows.
-410	Query INTERRUPTED	The next command was received before transmitting all of the requested responses. The response was interrupted, and the output buffer was cleared.
-420	Query UNTERMINATED	The response was requested but impossible because the received query was incomplete. The output buffer was cleared.
-430	Query DEADLOCKED	Processing cannot be continued due to full buffer. The output buffer was cleared.
-440	Query UNTERMINATED after indefinite response	There is another query after the query that requests arbitrary length response. No response is transmitted for the subsequent query.

Errors in remote control are recorded in the error queue, and they can be read, one by one starting from oldest error, with the query :SYSTem:ERRor?. After all errors were read, if the error queue is further read, +0,“No error” will be returned. The error queue can be cleared with *CLS command.
The data remaining in the input buffer or output buffer due to occurrence of a problem can be cleared with the device clear (DCL, SDC), which is one of the interface messages.

Other errors may occur depending on the situation. In this case, see the error message for the outline of the error.

For the items that can be operated from the panel, same error messages as those for panel operation are displayed. Accordingly, check the description of panel operations related to each command and query. The errors that occur in usual measurement are also displayed under the remote control.

NOTES

- Reproduction of the instruction manual, part or whole, is forbidden without prior written permission.
- The contents of the instruction manual are subject to change without notice.
- Information provided in the instruction manual is intended to be accurate and reliable. However, we assume no responsibility for any damage regarding the contents of the instruction manual.
  If you have any uncertainty or you found an error or omission, please contact NF Corporation or one of our representatives from which you purchased the product.

ZM2376 Instruction Manual (Remote Control)

NF Corporation

6-3-20 Tsunashima Higashi, Kohoku-ku, Yokohama
223-8508, JAPAN
Phone: +81-45-545-8128 Fax: +81-45-545-8187
http://www.nfcorp.co.jp/

http://www.nfcorp.co.jp/

NF Corporation

6-3-20 Tsunashima Higashi, Kohoku-ku, Yokohama 223-8508 JAPAN