



**RIGOL**

# DM858 Series

## Digital Multimeter

### Programming Guide

Feb. 2024



## **Guaranty and Declaration**

### **Copyright**

© 2024 RIGOL TECHNOLOGIES CO., LTD. All Rights Reserved.

### **Trademark Information**

RIGOL® is the trademark of RIGOL TECHNOLOGIES CO., LTD.

### **Notices**

- RIGOL products are covered by P.R.C. and foreign patents, issued and pending.
- RIGOL reserves the right to modify or change parts of or all the specifications and pricing policies at the company's sole decision.
- Information in this publication replaces all previously released materials.
- Information in this publication is subject to change without notice.
- RIGOL shall not be liable for either incidental or consequential losses in connection with the furnishing, use, or performance of this manual, as well as any information contained.
- Any part of this document is forbidden to be copied, photocopied, or rearranged without prior written approval of RIGOL.

### **Product Certification**

RIGOL guarantees that this product conforms to the national and industrial standards in China as well as the ISO9001:2015 standard and the ISO14001:2015 standard. Other international standard conformance certifications are in progress.

### **Contact Us**

If you have any problem or requirement when using our products or this manual, please contact RIGOL.

E-mail: [service@rigol.com](mailto:service@rigol.com)

Website: <http://www.rigol.com>

Section	Description	Page
	List of Figures.....	VII
	List of Tables.....	VIII
1	Document Overview .....	1
2	Programming Overview .....	3
2.1	SCPI Introduction .....	3
2.2	Remote Control .....	5
2.2.1	Remote Control via USB .....	6
2.2.2	Remote Control via LAN .....	6
3	Command System .....	8
3.1	ABORt .....	8
3.2	FETCh? .....	8
3.3	INITiate[:IMMediate] .....	9
3.4	OUTPut:TRIGger:SLOPe .....	10
3.5	R? .....	10
3.6	READ? .....	11
3.7	SAMPle:COUNT .....	12
3.8	UNIT:TEMPerature .....	12
3.9	CALCulate Commands .....	13
3.9.1	CALCulate:AVERage:ALL? .....	14
3.9.2	CALCulate:AVERage:AVERage? .....	14
3.9.3	CALCulate:AVERage:CLEar[:IMMediate] .....	15
3.9.4	CALCulate:AVERage:COUNT? .....	15
3.9.5	CALCulate:AVERage:MAXimum? .....	16
3.9.6	CALCulate:AVERage:MINimum? .....	16
3.9.7	CALCulate:AVERage:SDEVIation? .....	17
3.9.8	CALCulate:AVERage[:STATe] .....	17
3.9.9	CALCulate:CLEar[:IMMediate] .....	18
3.9.10	CALCulate:LIMit:CLEar[:IMMediate] .....	18
3.9.11	CALCulate:LIMit:LOWer[:DATA] .....	19
3.9.12	CALCulate:LIMit[:STATe] .....	20
3.9.13	CALCulate:LIMit:UPPer[:DATA] .....	20
3.9.14	CALCulate:SCALE:DB:REFerence .....	21
3.9.15	CALCulate:SCALE:DBM:REFerence .....	22

3.9.16	CALCulate:SCALE:FUNCTion	22
3.9.17	CALCulate:SCALE[:STATe]	23
3.10	CONFigure Commands	24
3.10.1	CONFigure:CAPacitance	24
3.10.2	CONFigure:CONTinuity	25
3.10.3	CONFigure:CURRent:AC	26
3.10.4	CONFigure:CURRent:DC	27
3.10.5	CONFigure:DIODE	28
3.10.6	CONFigure:FREQuency	28
3.10.7	CONFigure:FRESistance	29
3.10.8	CONFigure:PERiod	29
3.10.9	CONFigure:RESistance	30
3.10.10	CONFigure:TEMPerature	31
3.10.11	CONFigure:VOLTage:AC	32
3.10.12	CONFigure:VOLTage:DC	33
3.10.13	CONFigure?	34
3.11	DATA Commands	34
3.11.1	DATA:LAST?	34
3.11.2	DATA:POINts?	35
3.11.3	DATA:POINts:EVENT:THReshold	35
3.11.4	DATA:REMOve?	36
3.12	HCOPY Commands	37
3.12.1	HCOPY:SDUMp:DATA?	37
3.12.2	HCOPY:SDUMp:DATA:FORMat	38
3.13	IEEE488.2 Common Commands	38
3.13.1	*CLS	40
3.13.2	*ESE	40
3.13.3	*ESR?	41
3.13.4	*IDN?	42
3.13.5	*OPC	42
3.13.6	*PSC	43
3.13.7	*RST	43
3.13.8	*SRE	44
3.13.9	*STB?	45
3.13.10	*TRG	45
3.14	LXI Commands	46
3.14.1	LXI:MDNS:ENABLE	46

3.14.2	LXI:MDNS:HNAME[:RESolved]?	46
3.14.3	LXI:MDNS:SNAME:DESired	47
3.14.4	LXI:MDNS:SNAME[:RESolved]?	47
3.14.5	LXI:RESet	48
3.14.6	LXI:REStart	48
3.15	MEASure Commands	49
3.15.1	MEASure:CAPacitance?	49
3.15.2	MEASure:CONTinuity?	50
3.15.3	MEASure:CURREnt:AC?	50
3.15.4	MEASure:CURREnt:DC?	51
3.15.5	MEASure:DIODE?	52
3.15.6	MEASure:FREQuency?	53
3.15.7	MEASure:FRESistance?	53
3.15.8	MEASure:PERiod?	54
3.15.9	MEASure:RESistance?	55
3.15.10	MEASure:TEMPerature?	56
3.15.11	MEASure:VOLTage:AC?	57
3.15.12	MEASure:VOLTage:DC?	58
3.16	MMEMory Commands	58
3.16.1	MMEMory:CATalog[:ALL]?	59
3.16.2	MMEMory:CDIRectory	59
3.16.3	MMEMory:COpy	60
3.16.4	MMEMory:DElete	61
3.16.5	MMEMory:LOAD:PREFerences	62
3.16.6	MMEMory:LOAD:STATe	62
3.16.7	MMEMory:MDIRectory	63
3.16.8	MMEMory:MOVE	63
3.16.9	MMEMory:RDIRectory	64
3.16.10	MMEMory:STORE:PREFerences	65
3.16.11	MMEMory:STORE:STATe	65
3.16.12	MMEMory:STORE:DATA	66
3.16.13	MMEMory:STATe:RECall:AUTO	66
3.17	SENSe Commands	67
3.17.1	[SENSe]:CAPacitance:NULL[:STATe]	67
3.17.2	[SENSe]:CAPacitance:NULL:VALue	68
3.17.3	[SENSe]:CAPacitance:NULL:VALue:AUTO	69
3.17.4	[SENSe]:CAPacitance:RANGe	69

3.17.5	[SENSe]:CAPacitance:RANGe:AUTO	70
3.17.6	[SENSe]:CAPacitance:SECondary	71
3.17.7	[SENSe]:CURRent:AC:NULL[:STATe]	71
3.17.8	[SENSe]:CURRent:AC:NULL:VALue	72
3.17.9	[SENSe]:CURRent:AC:NULL:VALue:AUTO	73
3.17.10	[SENSe]:CURRent:AC:RANGe	73
3.17.11	[SENSe]:CURRent:AC:RANGe:AUTO	74
3.17.12	[SENSe]:CURRent:AC:SECondary	75
3.17.13	[SENSe]:CURRent:DC:NULL[:STATe]	76
3.17.14	[SENSe]:CURRent:DC:NULL:VALue	76
3.17.15	[SENSe]:CURRent:DC:NULL:VALue:AUTO	77
3.17.16	[SENSe]:CURRent:DC:RANGe	78
3.17.17	[SENSe]:CURRent:DC:RANGe:AUTO	78
3.17.18	[SENSe]:CURRent[:DC]:NPLC	79
3.17.19	[SENSe]:CURRent[:DC]:RESolution	80
3.17.20	[SENSe]:CURRent[:DC]:SECondary	80
3.17.21	[SENSe]:DATA2?	81
3.17.22	[SENSe]:DATA2:CLear[:IMMEDIATE]	82
3.17.23	[SENSe]:FREQUency:NULL[:STATe]	82
3.17.24	[SENSe]:FREQUency:NULL:VALue	83
3.17.25	[SENSe]:FREQUency:NULL:VALue:AUTO	83
3.17.26	[SENSe]:FREQUency:VOLTagE:RANGe	84
3.17.27	[SENSe]:FREQUency:VOLTagE:RANGe:AUTO	85
3.17.28	[SENSe]:FREQUency:SECondary	86
3.17.29	[SENSe]:FRESistance:NULL[:STATe]	86
3.17.30	[SENSe]:FRESistance:NULL:VALue	87
3.17.31	[SENSe]:FRESistance:NULL:VALue:AUTO	88
3.17.32	[SENSe]:FRESistance:NPLC	88
3.17.33	[SENSe]:FRESistance:RANGe	89
3.17.34	[SENSe]:FRESistance:RANGe:AUTO	90
3.17.35	[SENSe]:FRESistance:RESolution	91
3.17.36	[SENSe]:FRESistance:SECondary	91
3.17.37	[SENSe]:FUNctIon	92
3.17.38	[SENSe]:PERiod:NULL[:STATe]	93
3.17.39	[SENSe]:PERiod:NULL:VALue	94
3.17.40	[SENSe]:PERiod:NULL:VALue:AUTO	94
3.17.41	[SENSe]:PERiod:VOLTagE:RANGe	95

3.17.42	[SENSe]:PERiod:VOLTagE:RANGe:AUTO	96
3.17.43	[SENSe]:PERiod:SECondary	96
3.17.44	[SENSe]:RESistance:NPLC	97
3.17.45	[SENSe]:RESistance:NULL[:STATe]	98
3.17.46	[SENSe]:RESistance:NULL:VALue	99
3.17.47	[SENSe]:RESistance:NULL:VALue:AUTO	99
3.17.48	[SENSe]:RESistance:RANGe	100
3.17.49	[SENSe]:RESistance:RANGe:AUTO	101
3.17.50	[SENSe]:RESistance:RESolution	101
3.17.51	[SENSe]:RESistance:SECondary	102
3.17.52	[SENSe]:VOLTagE:AC:NULL[:STATe]	103
3.17.53	[SENSe]:VOLTagE:AC:NULL:VALue	103
3.17.54	[SENSe]:VOLTagE:AC:NULL:VALue:AUTO	104
3.17.55	[SENSe]:VOLTagE:AC:RANGe	105
3.17.56	[SENSe]:VOLTagE:AC:RANGe:AUTO	106
3.17.57	[SENSe]:VOLTagE:AC:SECondary	106
3.17.58	[SENSe]:VOLTagE:DC:NULL[:STATe]	107
3.17.59	[SENSe]:VOLTagE:DC:NULL:VALue	108
3.17.60	[SENSe]:VOLTagE:DC:NULL:VALue:AUTO	108
3.17.61	[SENSe]:VOLTagE:DC:RANGe	109
3.17.62	[SENSe]:VOLTagE:DC:RANGe:AUTO	110
3.17.63	[SENSe]:VOLTagE[:DC]:NPLC	110
3.17.64	[SENSe]:VOLTagE[:DC]:RESolution	111
3.17.65	[SENSe]:VOLTagE[:DC]:SECondary	112
3.18	STATus Commands	113
3.18.1	STATus:OPERation:CONDition?	116
3.18.2	STATus:OPERation:ENABLE	116
3.18.3	STATus:OPERation[:EVENT]?	117
3.18.4	STATus:PRESet	118
3.18.5	STATus:QUEStionable:CONDition?	118
3.18.6	STATus:QUEStionable:ENABLE	119
3.18.7	STATus:QUEStionable[:EVENT]?	119
3.19	SYSTem Commands	120
3.19.1	SYSTem:BEEPer[:IMMEdiate]	120
3.19.2	SYSTem:BEEPer:STATe	121
3.19.3	SYSTem:COMMunicate:LAN:AUTOip	121
3.19.4	SYSTem:COMMunicate:LAN:CONTRol?	122

3.19.5	SYSTem:COMMunicate:LAN:DHCP	123
3.19.6	SYSTem:COMMunicate:LAN:DNS	123
3.19.7	SYSTem:COMMunicate:LAN:GATeway	124
3.19.8	SYSTem:COMMunicate:LAN:HOSTname	125
3.19.9	SYSTem:COMMunicate:LAN:IPADdress	126
3.19.10	SYSTem:COMMunicate:LAN:MAC?	126
3.19.11	SYSTem:COMMunicate:LAN:MANuip	127
3.19.12	SYSTem:COMMunicate:LAN:SMASK	128
3.19.13	SYSTem:COMMunicate:LAN:UPDate	128
3.19.14	SYSTem:DATE	129
3.19.15	SYSTem:ERRor?	130
3.19.16	SYSTem:TIME	130
3.19.17	SYSTem:VERSion?	131
3.20	TRIGger Commands	132
3.20.1	TRIGger:COUNt	132
3.20.2	TRIGger:SOURce	132
4	Programming Examples	134
4.1	Programming Preparations	134
4.2	LabVIEW Programming Example	134
4.3	Visual Basic Programming Example	139
4.4	Visual C++ Programming Example	142



# List of Figures

Figure 3.1 DM858/DM858E Status System ..... 113

# List of Tables

Table 3.5 Math Operation .....	13
Table 3.14 Resolution, Measurement Speed, and Integration Time .....	24
Table 3.26 Table of the Bit Definition of Standard Event Status Register .....	38
Table 3.27 Table of the Bit Definition of Status Byte Register .....	39
Table 3.118 Bit Definitions for the Questionable Data Register .....	114
Table 3.119 Bit Definitions for the Operation Status Register .....	115

# 1 Document Overview

This manual introduces how to program and control DM858 series digital multimeter via the remote interface by using the SCPI (Standard Commands for Programmable Instruments) commands. DM858 series digital multimeter can communicate with PC through the USB or LAN interface.



## TIP

For the latest version of this manual, download it from RIGOL official website (<http://www.rigol.com>).

## Publication Number

PGC11101-1110


## Software Version

00.01.00

Software upgrade might change or add product features. Please acquire the latest version of the manual from RIGOL website or contact RIGOL to upgrade the software.

## Format Conventions in this Manual

### 1. Key


The front panel key is denoted by the menu key icon. For example,  indicates the "Trig" key.

### 2. Menu

The menu item is denoted by the format of "Menu Name (Bold) + Character Shading" in the manual. For example, **Measure** indicates the "Measure" menu item. You can click or tap **Measure** to access the "Measure" menu.

### 3. Operation Procedures

The next step of the operation is denoted by ">" in the manual. For example, 

> **Storage** indicates first clicking or tapping  and then clicking or tapping **Storage**.

### 4. Connector

The front/rear panel connector is denoted by "Brackets + Connector Name (Bold)", for example, [EXT TRIG].

## Content Conventions in this Manual

DM858 series digital multimeter includes DM858 and DM858E models. The table below shows the main differences between the two models. Their operation methods

are the same. Unless otherwise specified, this manual takes DM858 as an example to illustrate the basic operation methods of DM858 series.

<b>Specifications for Different Models</b>		
	DM858	DM858E
DCV Accuracy (1 year)	0.030%	0.060%
Max. Reading Rate	125 readings/s	80 readings/s
No. of Points in Data Logging Memory	500,000 data points	20,000 data points
Current Measurement Range	100 $\mu$ A to 10 A	100 $\mu$ A to 3 A
Capacitance Measurement Range	1 nF to 10 mF	1 nF to 1 mF

## 2 Programming Overview

### 2.1 SCPI Introduction

SCPI (Standard Commands for Programmable Instruments) is a standardized instrument programming language that is built upon the existing standard IEEE 488.1 and IEEE 488.2 and conforms to various standards, such as the floating point operation rule in IEEE 754 standard, ISO 646 7-bit coded character set for information interchange (equivalent to ASCII programming). The SCPI commands provide a hierarchical tree structure, and consist of multiple subsystems. Each command subsystem consists of one root keyword and one or more sub-keywords.

#### Syntax

The keywords are separated by colons, and following the keywords are the parameter settings available. The command ending with a question mark indicates querying a certain function and returns the query results. The keywords of the command and the first parameter are separated by a space.

For example,

```
SYSTem:BEEPer:STATe <bool>
```

```
SYSTem:BEEPer:STATe?
```

**SYSTem** is the root keyword of the command, **BEEPer** is the second-level keyword, and **STATe** is the third-level keyword. Keyword at different levels are separated by colons. *<bool>* indicates the parameter that can be set. The command ending with a quotation mark indicates querying a function. The command keywords **SYSTem:BEEPer:STATe** and the parameter *<bool>* are separated by a space.

In some commands with parameters, "," is often used to separate multiple parameters. For example,

```
SYSTem:DATE <year>,<month>,<day>
```

#### Symbol Description

The following symbols are not sent with the commands.

##### 1. Braces { }

The contents in the braces can contain one or multiple parameters. These parameters can be omitted or used for several times. Parameters are usually separated by the vertical bar "|". When using the command, you must select one of the parameters.

##### 2. Vertical Bar |

The vertical bar is used to separate multiple parameters. When using the command, you must select one of the parameters.

### 3. Square Brackets [ ]

The contents in the square brackets can be omitted.

### 4. Angle Brackets < >

The parameter enclosed in the angle brackets must be replaced by an effective value.

## Parameter Type

### 1. Bool

The parameter can be set to ON, OFF, 1, or 0. For example,

```
SYSTem: BEEPer: STATe <bool>
```

```
SYSTem: BEEPer: STATe?
```

Wherein, <bool> can be set to {1|ON|0|OFF}. The query returns 1 or 0.

### 2. Discrete

The parameter can be any of the values listed. For example,

```
OUTPut: TRIGger: SLOPe <slope>
```

```
OUTPut: TRIGger: SLOPe?
```

Wherein,

- <slope> can be set to POSitive|NEGative.
- The query returns an abbreviated form: POS or NEG.

### 3. Integer

Unless otherwise specified, the parameter can be any integer (NR1 format) within the effective value range.



#### CAUTION

**Do not set the parameter to a decimal; otherwise, errors may occur.**

For example,

```
SAMPle: COUNT <sample>
```

Wherein, <sample> can be set to an integer ranging from 1 to 2000. The query returns an integer ranging from 1 to 2000.

### 4. Real

The parameter can be any real number within the effective value range, and this command accepts parameter input in decimal (NR2 format) and scientific notation (NR3 format). For example,

```
[SENSe]:FREQuency:NULL:VALue {<value>|<lim>}
```

```
[SENSe]:FREQuency:NULL:VALue?
```

Wherein, <value> can be set to any real number ranging from -1.2E+3 (-120 kHz) to +1.2E+3 (+120 kHz). The query returns a real number in scientific notation.

## 5. ASCII String

The parameter can be the combinations of ASCII characters. Strings must begin and end with paired double quotation marks, such as:

```
SYSTem:COMMunicate:LAN:GATEway "<string>"
```

Wherein, "<string>" can be set to "192.168.1.1".

### Use the MIN, MAX, and DEF parameters

For some commands, you can substitute MIN or MAX in place of a parameter. In some cases, you can also substitute DEF. For example,

```
[SENSe]:CAPacitance:RANGe {<range>|<lim>}
```

Wherein, <lim> can be set to MIN|MAX|DEF.

Instead of selecting a specific value for <range>, you can substitute MIN to set the range to its minimum value and MAX to set the range to its maximum value. You can also use DEF to set the range to its default value.

### Command Abbreviation

All the commands are case-insensitive. They can all be in upper case or in lower case. If abbreviation is used, you must input all the capital letters in the command. For example,

```
SYSTem:BEEPer:STATe?
```

can be abbreviated as

```
SYST:BEEP:STAT?
```

## 2.2 Remote Control

This instrument can be connected to the PC via the USB and LAN interfaces to set up communication and realize remote control through the PC. The remote control can be realized by using SCPI (Standard Commands for Programmable Instruments) commands.

### PC Software

Users usually need to use the PC software to send commands to control the instrument remotely. RIGOL Ultra Sigma is recommended. When the instrument is connected to the PC via the USB or LAN interface, the Ultra Sigma software can search for instrument resources and enable command interaction.

Log in to the RIGOL official website. Click Support and select Soft/Firmware to obtain the Ultra Sigma software package and help documentation.

### Web Control

When the instrument is connected to the PC via the LAN interface, you can use Web Control to send SCPI commands from the PC to the instrument. Operation

Procedures:

1. Obtain the instrument's IP address and input it in the browser address bar to log in to the Web Control page.
2. After you enter the Web Control interface, click the "SCPI Panel Control" button to enter the SCPI Command interface.
3. Input the specified SCPI command and then click **Send & Read** to send the command. The operation process and the returned value will be displayed in the current interface.

## 2.2.1 Remote Control via USB

### 1. Connect the device

Use the USB cable to connect the rear-panel USB DEVICE interface of the instrument to the USB HOST interface of the PC.

### 2. Search for the device resource

Start up Ultra Sigma and the software will automatically search for the resource currently connected to the PC via the USB interface. You can also click **USB-TMC** to search for the resource.

### 3. View the device resource

The resources found will appear under the "RIGOL Online Resource" directory, and the model number and USB interface information of the instrument will also be displayed.

### 4. Control the instrument remotely

Right-click the device resource name and select "SCPI Panel Control" to open the remotely command control panel. Then you can send commands and read data through the panel. For details about the SCPI commands and programming, refer to the Programming Guide of this instrument.

## 2.2.2 Remote Control via LAN

### 1. Connect the device

Use the network cable to connect the instrument to your local area network (LAN).

### 2. Configure network parameters

Configure the network parameters of the instrument in **Utility>IO** menu.



Click or tap the Notification Area at the lower-right corner of the screen, then the **Utility** menu is displayed. Click or tap **IO**, and then click or tap the input field of **GPIB** to input the GPIB address with the pop-up numeric keypad.

### 3. Search for Search device resource

Start up Ultra Sigma and click **LAN** to open the panel as shown in the figure below. Click **Search** and the software searches for the instrument resources currently connected to the LAN and the resources found are displayed at the right section of the window as shown in the figure below. Click **OK** to add it.



Besides, you can input the IP address of the instrument manually into the text field under "Manual Input LAN Instrument IP", then click **TEST**. If the instrument passes the test, click **Add** to add the instrument to the LAN instrument resource list in the right section; if the instrument fails the test, please check whether the IP address that you input is correct, or use the auto search method to add the instrument resource.

### 4. View the device resource

The resources found will appear under the "RIGOL Online Resource" directory.

### 5. Control the instrument remotely

Right-click the device resource name and select "SCPI Panel Control" to open the remotely command control panel. Then you can send commands and read data through the panel.

### 6. Load LXI webpage

As this instrument conforms to LXI CORE 2011 DEVICE standards, you can load LXI web page through Ultra Sigma (right-click the instrument resource name and select "LXI-Web"). Various important information about the instrument (including the model, manufacturer, serial number, description, MAC address, and IP address) will be displayed on the web page. You can also directly input the IP address of the instrument in the address bar of the PC browser to load the LXI web page.

## 3 Command System

---

This chapter introduces the syntax, functions, parameters, and usage of each command for DM858 series Digital Multimeter.



### NOTE

In this manual, unless otherwise noted, commands related to instrument parameter settings cannot be set with unit. In addition, the descriptions of the parameter ranges and default values in this manual are based on the current software version (see *\*IDN?*) and are subject to change without notice.

### 3.1 ABORt

---

#### Syntax

ABORt

#### Description

Aborts a measurement in progress, returning the instrument to the trigger idle state.

#### Parameter

None.

#### Remarks

This command can abort a measurement when the instrument is waiting for a trigger or perform one or a series of prolonged measurements.

#### Return Format

None.

#### Examples

```
ABORt /*Aborts a measurement in progress.*/
```

### 3.2 FETCh?

---

#### Syntax

FETCh?

#### Description

Waits for the measurements to complete and transfers the reading from memory to the instrument's output buffer.

**Parameter**

None.

**Remarks**

This command reads the readings from the reading memory without clearing the memory. DM858 can store up to 500,000 readings while DM858E can store up to 20,000 readings. If the memory overflows, the new measurement result will overwrite the oldest stored measurement result.

**Return Format**

The query returns the reading. If the memory has no reading, there is no returned value.

**Examples**

```
FETCh? /*The query returns the measurement reading.*/
```

## 3.3 INITiate[:IMMediate]

**Syntax**

```
INITiate[:IMMediate]
```

**Description**

Changes the trigger state from the "idle" state to the "wait-for-trigger" state. Measurements will begin when the specified trigger conditions are satisfied.

**Parameter**

None.

**Remarks**

- After the measurement begins, the readings will be stored in the reading memory of the instrument.
- DM858 can store up to 500,000 readings while DM858E can store up to 20,000 readings. If the memory overflows, the new measurement result will overwrite the oldest stored measurement result.
- To read measurement results from the reading memory, refer to *FETCh?*. Using *DATA:REMove?* or *R?* to read data will clear all or part of the results.

**Return Format**

None.

**Examples**

```
INITiate:IMMediate /*Changes the trigger state from the "idle"
state to the "wait-for-trigger" state. Measurements will begin when
the specified trigger conditions are satisfied.*/
```

## 3.4 OUTPut:TRIGger:SLOPe

**Syntax**

```
OUTPut:TRIGger:SLOPe <polar>
```

```
OUTPut:TRIGger:SLOPe?
```

**Description**

Sets or queries the output polarity of VMC.

**Parameter**

Name	Type	Range	Default
<polar>	Discrete	{POSitive NEGative}	POSitive

**Remarks**

When the Output function is enabled, this command sets the polarity of the output signal of the rear-panel **[VM COMP]** connector after each measurement.

- **POSitive:** sets the polarity to positive.
- **NEGative:** sets the polarity to negative.

**Return Format**

The query returns POS or NEG.

**Examples**

```
OUTPut:TRIGger:SLOPe POSitive /*Sets the VMC output polarity to
positive.*/
OUTPut:TRIGger:SLOPe? /*Queries the output polarity of VMC. The
query returns POS.*/
```

## 3.5 R?

**Syntax**

```
R? [<max_readings>]
```

**Description**

Reads and removes the specified measurement results from the reading memory (starting from the earliest results).

**Parameter**

Name	Type	Range	Default
<max_readings>	Integer	Refer to <i>Remarks</i>	Refer to <i>Remarks</i>

**Remarks**

- The range of [<max\_readings>] is related to the instrument model. For DM858, it ranges from 1 to 500,000; for DM858E, it ranges from 1 to 20,000. If [<max\_readings>] is omitted, it is interpreted as reading and removing all measurement results.
- The number of readings returned may be less than the required number, which depends on the number of readings stored in the reading memory of the instrument.

**Return Format**

The query returns a string. For example, the query might return #247-1.63969181E+01,-2.81863565E+01,-3.03502037E+01; wherein, "#2" means that the last 2 digits indicate how many characters are contained in the returned memory string while the "47" following "#2" means that th string has 47 remaining digits.

**Examples**

```
R? 3 /*Reads and removes the three earliest readings. The query
returns #247-1.63969181E+01,-2.81863565E+01,-3.03502037E+01.*/
```

## 3.6 READ?

**Syntax**

READ?

**Description**

Begins a new group of measurements. Waits for all measurements to complete and sends all available results.

**Parameter**

None.

**Remarks**

The multimeter will begin measurements when the specified trigger conditions are satisfied following the receipt of this command. The measurement reading is then sent to volatile memory and the instrument's output buffer.

**Return Format**

The query returns one or a group of measurement results.

**Examples**

```
READ? /*Sends measurement results from reading memory. The query
returns
-4.98748741E-01,-4.35163427E-01,-4.33118686E-01,-3.48109378E-01.*/
```

## 3.7 SAMPLE:COUNT

**Syntax**

**SAMPLE:COUNT** {<value>|<lim>}

**SAMPLE:COUNT?**

**Description**

Sets or queries the number of samples (measurements) the instrument will take per trigger in Single trigger mode.

**Parameter**

Name	Type	Range	Default
<value>	Integer	1 to 2000	1
<lim>	Discrete	{MIN MAX DEF}	-

**Remarks**

- The set sample count is valid only when the trigger source (*TRIGger:SOURce*) is set to single trigger.
- You can also specify a trigger count (*TRIGger:COUNt*). At this moment, the number of measurements for single trigger is the product of the sample count per trigger and trigger count.

**Return Format**

The query returns an integer.

**Examples**

```
SAMPLE:COUNT 200 /*Sets the sample count per trigger to 200.*/
SAMPLE:COUNT? /*Queries the sample count per trigger. The query
returns 200.*/
```

## 3.8 UNIT:TEMPerature

**Syntax**

**UNIT:TEMPerature** <unit>

**UNIT:TEMPerature?**

### Description

Sets or queries the units to be used for temperature measurements.

### Parameter

Name	Type	Range	Default
<unit>	Discrete	{C F K}	C

### Remarks

- **C:** °C
- **F:** °F
- **K:** K

### Return Format

The query returns C, F, or K.

### Examples

```
UNIT:TEMPerature F /*Sets the unit for temperature measurements to °F.*/
UNIT:TEMPerature? /*Queries the unit for temperature measurements. The query returns F.*/
```

## 3.9 CALCulate Commands

**CALCulate** commands are used to set parameters for statistics, dB, dBm, and limit operations. The table below shows the math operations that can be performed by different measurement functions. To set relative operation, use *SENSe Commands*.

**Table 3.5 Math Operation**

Measurement Function	Available Math Operation
DCV/ACV	Statistics, Limit, dBm, dB, Relative
DCI/ACI	Statistics, Limit, Relative
2WR/4WR	Statistics, Limit, Relative
CAP	Statistics, Limit, Relative
SENSOR	Statistics, Limit, Relative
FREQ/PREIOD	Statistics, Limit, Relative

Measurement Function	Available Math Operation
CONT/DIODE	NA

### 3.9.1 CALCulate:AVERage:ALL?

#### Syntax

**CALCulate:AVERage:ALL?**

#### Description

Queries the average value, standard deviation, minimum value, and maximum value for the Statistics operation.

#### Parameter

None.

#### Remarks

This command is valid only when the Statistics operation (*CALCulate:AVERage[:STATe]*) is enabled.

#### Return Format

The query returns a string. For example, the query might return -6.60019915E+01,3.12397977E-04,-6.60040000E+01,-6.60010000E+01, indicating that the average value is -6.60019915E+01, the standard deviation is 3.12397977E-04, the minimum value is -6.60040000E+01, and the maximum value is -6.60010000E+01.

#### Examples

```
CALCulate:AVERage:ALL? /*Queries the average value, standard
deviation, minimum value, and maximum value for the Statistics
operation. The query returns -6.60019915E
+01,3.12397977E-04,-6.60040000E+01,-6.60010000E+01.*/
```

### 3.9.2 CALCulate:AVERage:AVERage?

#### Syntax

**CALCulate:AVERage:AVERage?**

#### Description

Queries the average value for the Statistic function.

#### Parameter

None.



**Remarks**

This command is valid only when the Statistics operation (*CALCulate:AVERage[:STATe]*) is enabled.

**Return Format**

The query returns the average value in scientific notation.

**Examples**

```
CALCulate:AVERage:AVERage? /*Queries the average value for the
Statistic function. The query returns 1.23450000E+01.*/
```

### 3.9.3 CALCulate:AVERage:CLEar[:IMMEDIATE]

**Syntax**

```
CALCulate:AVERage:CLEar[:IMMEDIATE]
```

**Description**

Clears all calculated statistics: minimum, maximum, average, count, and standard deviation.

**Parameter**

None.

**Remarks**

- It does not clear the measurement results in the reading memory.
- To clear statistics, limits, histogram data, and measurement data, use *CALCulate:CLEar[:IMMEDIATE]*.

**Return Format**

None.

**Examples**

None.

### 3.9.4 CALCulate:AVERage:COUNT?

**Syntax**

```
CALCulate:AVERage:COUNT?
```

**Description**

Queries the number of readings for the Statistical operation.

**Parameter**

None.

**Remarks**

This command is valid only when the Statistics operation (*CALCulate:AVERage[:STATe]*) is enabled.

**Return Format**

The query returns an integer.

**Examples**

```
CALCulate:AVERage:COUNT? /*Queries the number of readings for the  
Statistical operation. The query returns 11986.*/
```

### 3.9.5 CALCulate:AVERage:MAXimum?

**Syntax**

```
CALCulate:AVERage:MAXimum?
```

**Description**

Queries the maximum value of the statistic measurement.

**Parameter**

None.

**Remarks**

This command is valid only when the Statistics operation (*CALCulate:AVERage[:STATe]*) is enabled.

**Return Format**

The query returns the maximum value in scientific notation.

**Examples**

```
CALCulate:AVERage:MAXimum? /*Queries the maximum value of the  
statistic measurement. The query returns -2.40000000E+01.*/
```

### 3.9.6 CALCulate:AVERage:MINimum?

**Syntax**

```
CALCulate:AVERage:MINimum?
```

**Description**

Queries the minimum value of the Statistic operation.

**Parameter**

None.

**Remarks**

This command is valid only when the Statistics operation (*CALCulate:AVERage[:STATe]*) is enabled.

**Return Format**

The query returns the minimum value in scientific notation.

**Examples**

```
CALCulate:AVERage:MINimum? /*Queries the minimum value of the
Statistic operation. The query returns -2.70000000E+01.*/
```

### 3.9.7 CALCulate:AVERage:SDEViation?

**Syntax**

```
CALCulate:AVERage:SDEViation?
```

**Description**

Queries the standard deviation for the Statistics operation.

**Parameter**

None.

**Remarks**

This command is valid only when the Statistics operation (*CALCulate:AVERage[:STATe]*) is enabled.

**Return Format**

The query returns the standard deviation in scientific notation.

**Examples**

```
CALCulate:AVERage:SDEViation? /*Queries the standard deviation for
the Statistics operation. The query returns 3.50353538E-04.*/
```

### 3.9.8 CALCulate:AVERage[:STATe]

**Syntax**

```
CALCulate:AVERage[:STATe] <bool>
```

```
CALCulate:AVERage[:STATe]?
```

**Description**

Sets or queries the on/off status of the Statistics operation.

**Parameter**

Name	Type	Range	Default
<bool>	Bool	{0 1 OFF ON}	0 OFF

**Remarks**

- This command is available only for measurement functions supporting the Statistics operation ([Table 3.5 Math Operation](#)).
- The Statistics operation is not available when the range is "auto range".

**Return Format**

The query returns 0 or 1.

**Examples**

```
CALCulate:AVERage ON /*Enables the Statistics operation.*/
CALCulate:AVERage? /*Queries the on/off status of the Statistics
operation. The query returns 1.*/
```

### 3.9.9 CALCulate:CLEar[:IMMediate]

**Syntax**

```
CALCulate:CLEar[:IMMediate]
```

**Description**

Clears all limit values, histogram data, statistical information, and measurement results.

**Parameter**

None.

**Remarks**

None.

**Return Format**

None.

**Examples**

```
None.
```

### 3.9.10 CALCulate:LIMit:CLEar[:IMMediate]

**Syntax**

```
CALCulate:LIMit:CLEar[:IMMediate]
```

**Description**

Clears the results of limit operation and clears bit 11 ("Lower Limit Failed") and bit 12 ("Upper Limit Failed") in the condition register of the Questionable Data register event register group.

**Parameter**

None.

**Remarks**

Executing this command does not clear the measurement results from the reading memory.

**Return Format**

None.

**Examples**

None.

### 3.9.11 CALCulate:LIMit:LOWer[:DATA]

**Syntax**

`CALCulate:LIMit:LOWer[:DATA] {<value>|<lim>}`

`CALCulate:LIMit:LOWer[:DATA]?`

**Description**

Sets or queries the lower limit for the current limit operation.

**Parameter**

Name	Type	Range	Default
<value>	Real	Refer to <i>Remarks</i>	0
<lim>	Discrete	{MIN MAX DEF}	-

**Remarks**

The range of <value> is -120% to +120% of the maximum measurement range. The unit of the limit value is related to the current measurement function. The lower limit cannot exceed the upper limit of the current setting (`CALCulate:LIMit:UPPer[:DATA]`).

**Return Format**

The query returns the lower limit for limit operation in scientific notation.

**Examples**

```
CALCulate:LIMit:LOWer:DATA 1 /*Sets the lower limit to 1 V for
limit operation.*/
CALCulate:LIMit:LOWer:DATA? /*Queries the lower limit for limit
operation. The query returns +1.00000000E+00.*/
```

**3.9.12 CALCulate:LIMit[:STATe]****Syntax**

```
CALCulate:LIMit[:STATe] <bool>
```

```
CALCulate:LIMit[:STATe]?
```

**Description**

Sets or queries the on/off status of the limit operation.

**Parameter**

Name	Type	Range	Default
<bool>	Bool	{0 1 OFF ON}	0 OFF

**Remarks**

- This command is available only for measurement functions supporting limit operation ([Table 3.5 Math Operation](#)).
- The limit operation is not available when the range is "auto range".

**Return Format**

The query returns 0 or 1.

**Examples**

```
CALCulate:LIMit:STATe ON /*Enables the limit operation.*/
CALCulate:LIMit:STATe? /*Queries the on/off status of the limit
operation. The query returns 1.*/
```

**3.9.13 CALCulate:LIMit:UPPer[:DATA]****Syntax**

```
CALCulate:LIMit:UPPer[:DATA] {<value>|<lim>}
```

```
CALCulate:LIMit:UPPer[:DATA]?
```

**Description**

Sets or queries the upper limit for the current limit operation.

**Parameter**

Name	Type	Range	Default
<value>	Real	Refer to <i>Remarks</i>	0
<lim>	Discrete	{MIN MAX DEF}	-

**Remarks**

The range of <value> is -120% to +120% of the maximum measurement range. The unit of the limit value is related to the current measurement function. The upper limit value cannot be less than the current lower limit (*CALCulate:LIMit:LOWerf:DATA*).

**Return Format**

The query returns the upper limit for limit operation in scientific notation.

**Examples**

```
CALCulate:LIMit:UPPer:DATA 2 /*Sets the upper limit to 2 V for the
current limit operation.*/
CALCulate:LIMit:UPPer:DATA? /*Queries the upper limit for the
current limit operation. The query returns +2.00000000E+00.*/
```

### 3.9.14 CALCulate:SCALE:DB:REference

**Syntax**

**CALCulate:SCALE:DB:REference** {<value>|<lim>}

**CALCulate:SCALE:DB:REference?**

**Description**

Sets or queries the dB relative value for dB operation.

**Parameter**

Name	Type	Range	Default
<value>	Real	-120 dBm to 120 dBm	0 dBm
<lim>	Discrete	{MIN MAX DEF}	-

**Remarks**

When dB operation is enabled, the multimeter calculates the dBm value and then subtracts the value from each voltage measurement result.

**Return Format**

The query returns the dB relative value in scientific notation, for example, 10.000000.

**Examples**

```
:CALCulate:SCALE:DB:REference 10 /*Sets the dB relative value to 10 dBm for dB operation.*/
:CALCulate:SCALE:DB:REference? /*Queries the dB relative value for dB operation. The query returns 10.000000.*/
```

**3.9.15 CALCulate:SCALE:DBM:REference****Syntax**

```
CALCulate:SCALE:DBM:REference {<value>|<lim>}
```

```
CALCulate:SCALE:DBM:REference?
```

**Description**

Sets or queries the dBm relative resistance.

**Parameter**

Name	Type	Range	Default
<value>	Real	2 $\Omega$ to 8000 $\Omega$	600 $\Omega$
<lim>	Discrete	{MIN MAX DEF}	-

**Remarks**

None.

**Return Format**

The query returns the dBm relative resistance, for example, 100.000000.

**Examples**

```
CALCulate:SCALE:DBM:REference 100 /*Sets the dBm relative resistance to 100  $\Omega$ .*/
CALCulate:SCALE:DBM:REference? /*Queries the dBm relative resistance. The query returns 100.000000.*/
```

**3.9.16 CALCulate:SCALE:FUNction****Syntax**

```
CALCulate:SCALE:FUNction <type>
```

```
CALCulate:SCALE:FUNction?
```

**Description**

Sets and queries the operation performed by the scaling function.



**Parameter**

Name	Type	Range	Default
<type>	Discrete	{DB DBM}	-

**Remarks**

- The scaling function supports the dB and dBm operations.
  - DB:** dB operation. The result is the difference between the input signal and the stored DB relative value (*CALCulate:SCALE:DB:REFERENCE*), with both values converted to dBm (dB = measurement result in dBm – relative value in dBm). dB scaling only applies to ACV and DCV measurements.
  - DBM:** dBm operation. The result is logarithmic and is based on a calculation of power delivered to a relative resistance (*CALCulate:SCALE:DBM:REFERENCE*), relative to 1 mW.  $Bm = 10 \times \log_{10}(\text{measurement result}^2 / \text{relative resistance} / 1 \text{ mW})$ . dBm scaling only applies to ACV and DCV measurements.
- After executing this command, use *CALCulate:SCALE[:STATe]* to initiate the selected operation.

**Return Format**

The query returns DB or DBM.

**Examples**

```
CALCulate:SCALE:FUNCTION DB /*Sets the operation performed by the
scaling function to dB operation.*/
CALCulate:SCALE:FUNCTION? /*Queries the operation performed by the
scaling function. The query returns DB.*/
```

### 3.9.17 CALCulate:SCALE[:STATe]

**Syntax**

*CALCulate:SCALE[:STATe]* <bool>

*CALCulate:SCALE[:STATe]?*

**Description**

Sets or queries the on/off status of the scaling function.

**Parameter**

Name	Type	Range	Default
<bool>	Bool	{0 1 OFF ON}	0 OFF

**Remarks**

- Scaling is off by default when you change measurement functions (for example, changing from DCV to ACV). You must re-enable scaling after changing measurement functions.
- This command is valid only in measurement functions for dB/dBm operation (see [Table 3.5 Math Operation](#)).
- Scaling cannot be enabled when the range is set to "auto range".
- You must execute `CALCulate:SCALE:FUNCTION` first and then enable the scaling function.

**Return Format**

The query returns 0 or 1.

**Examples**

```
:CALCulate:SCALE:STATE ON /*Enables the scaling function.*/
:CALCulate:SCALE:STATE? /*Queries the on/off status of the scaling
function. The query returns 1.*/
```

## 3.10 CONFigure Commands

**CONFigure** commands are used to set the instrument's basic measurement functions as well as range and resolution parameters. Use V, A, Hz,  $\Omega$  as the default units to specify the range (<range>) and resolution (<resolution>). The table below shows the relationships among the resolution, measurement speed, and integration time.

**Table 3.14 Resolution, Measurement Speed, and Integration Time**

Resolution	Measurement Speed	Integration Time
1000 ppm x range	Fast	0.4 PLC
100 ppm x range	Medium	5 PLC
10 ppm x range	Slow	20 PLC

**NOTE**

1 ppm = 0.000001, indicating one in a million; 1 PLC = 0.02 s.



### 3.10.1 CONFigure:CAPacitance

**Syntax**

`CONFigure:CAPacitance [{<range>|<lim>}]`

**Description**

Presets the multimeter with the specified range for capacitance measurement.

**Parameter**

Name	Type	Range	Default
<range>	Discrete	{1nF 10nF 100nF 1μF 10μF 100μF 1mF 10mF AUTO}	AUTO
<lim>	Discrete	{MIN MAX DEF}	-

**Remarks**

- Only DM858 supports the 10 mF range. For DM858E, the maximum capacitance range is 1 mF.
- The default unit of <range> is F. You can set the value in scientific notation, for example, 1E-6 (1 μF). If the parameter is omitted, it is interpreted as the default value.
- AUTO (auto range) automatically selects the range according to the input signal.
- This command only specifies the parameters without initiating the measurement.

**Return Format**

None.

**Examples**

```
CONFigure:CAPacitance 1E-9 /*Sets the range to 1 nF for capacitance
measurement.*/
CONFigure? /*Queries the current configuration. The query returns
CAP 1.00000000E-09.*/
```

## 3.10.2 CONFigure:CONTinuity

**Syntax**

**CONFigure:CONTinuity**

**Description**

Sets the continuity measurement parameters and trigger parameters to their default values for continuity measurement.

**Parameter**

None.

**Remarks**

This command only presets the parameters without initiating the measurement.

**Return Format**

None.

**Examples**

```

CONFigure:CONTInuity /*Presets the multimeter for continuity
measurement.*/
CONFigure? /*Queries the current configuration. The query returns
CONT.*/

```

**3.10.3 CONFigure:CURRent:AC****Syntax**

```
CONFigure:CURRent:AC [{<range>|<lim>}[, {<resolution>|<lim>}]]
```

**Description**

Presets the multimeter with the specified range and resolution for AC current measurement.

**Parameter**

Name	Type	Range	Default
<range>	Discrete	{100μA 1mA 10mA 100mA 1A 3A 10A AUTO}	AUTO
<lim>	Discrete	{MIN MAX DEF}	-
<resolution>	Real	Refer to <i>Table 3.14 Resolution, Measurement Speed, and Integration Time</i>	10 ppm x range

**Remarks**

- For <range>, 3 A range is only available for DM858E while 10 A range is only available for DM858.
- The default unit of <range> and <resolution> is A. You can set the parameters in scientific notation, for example, 1E-3 (0.001 A). If the parameter is omitted, it is interpreted as the default value.
- AUTO (auto range) automatically selects the range according to the input signal.
- This command only specifies the parameters without initiating the measurement.

**Return Format**

None.

## Examples

```
CONFigure:CURRent:AC 1,1E-5 /*Sets the range to 1 A and the
resolution to 10 µA for AC current measurement.*/
CONFigure? /*Queries the current configuration. The query returns
CURR:AC 1.00000000E+00,1.00000000E-05.*/
```

## 3.10.4 CONFigure:CURRent:DC

### Syntax

```
CONFigure:CURRent:DC [{<range>|<lim>}|{<resolution>|<lim>}]
```

### Description

Presets the multimeter with the specified range and resolution for DC current measurement.

### Parameter

Name	Type	Range	Default
<range>	Discrete	{100µA 1mA 10mA 100mA 1A 3A 10A AUTO}	AUTO
<lim>	Discrete	{MIN MAX DEF}	-
<resolution>	Real	Refer to <i>Table 3.14 Resolution, Measurement Speed, and Integration Time</i>	10 ppm x range

### Remarks

- For <range>, 3 A range is only available for DM858E while 10 A range is only available for DM858.
- The default unit of <range> and <resolution> is A. You can set the parameters in scientific notation, for example, 1E-3 (0.001 A). If the parameter is omitted, it is interpreted as the default value.
- AUTO (auto range) automatically selects the range according to the input signal.
- This command only specifies the parameters without initiating the measurement.

### Return Format

None.

### Examples

```
CONFigure:CURRent:DC 1,1E-5 /*Sets the range to 1 A and the
resolution to 10 µA for DC current measurement.*/
CONFigure? /*Queries the current configuration. The query returns
CURR 1.00000000E+00,1.00000000E-05.*/
```

### 3.10.5 CONFigure:DIODe

---

#### Syntax

CONFigure:DIODe

#### Description

Sets the diode measurement parameters and trigger parameters to their default values for diode measurement.

#### Parameter

None.

#### Remarks

This command only presets the parameters without initiating the measurement.

#### Return Format

None.

#### Examples

```
CONFigure:DIODe /*Presets the multimeter for diode measurement.*/  
CONFigure? /*Queries the current configuration. The query returns  
DIOD.*/
```

### 3.10.6 CONFigure:FREQuency

---

#### Syntax

CONFigure:FREQuency

#### Description

Sets the frequency measurement parameters and trigger parameters to their default values for frequency measurement.

#### Parameter

None.

#### Remarks

This command only presets the parameters without initiating the measurement.

#### Return Format

None.

#### Examples

```
CONFigure:FREQuency /*Presets the multimeter for frequency  
measurement.*/  
CONFigure? /*Queries the current configuration. The query returns  
FREQ.*/
```

### 3.10.7 CONFigure:FRESistance

#### Syntax

```
CONFigure:FRESistance [{<range>|<lim>}][,<resolution>|<lim>}]
```

#### Description

Presets the multimeter with the specified range and resolution for 4-wire resistance measurement.

#### Parameter

Name	Type	Range	Default
<range>	Discrete	{100Ω 1kΩ 10kΩ 100kΩ 1MΩ 10MΩ 50MΩ AUTO}	AUTO
<lim>	Discrete	{MIN MAX DEF}	-
<resolution>	Real	Refer to <i>Table 3.14 Resolution, Measurement Speed, and Integration Time</i>	10 ppm x range

#### Remarks

- The default unit of <range> and <resolution> is Ω. You can set the parameters in scientific notation, for example, 1E-3 (0.001 Ω). If the parameter is omitted, it is interpreted as the default value.
- AUTO (auto range) automatically selects the range according to the input signal.
- This command only specifies the parameters without initiating the measurement.

#### Return Format

None.

#### Examples

```
CONFigure:FRESistance 1000,1E-2 /*Sets the range to 1 kΩ and the resolution to 10 mΩ for 4-wire resistance measurement.*/
CONFigure? /*Queries the current configuration. The query returns FRES 1.00000000E+03,1.00000000E-02.*/
```

### 3.10.8 CONFigure:PERiod

#### Syntax

```
CONFigure:PERiod
```

**Description**

Sets period measurement parameters and trigger parameters to their default values for period measurement.

**Parameter**

None.

**Remarks**

This command only presets the parameters without initiating the measurement.

**Return Format**

None.

**Examples**

```
CONFigure:PERiod /*Presets the multimeter for period measurement.*/
CONFigure? /*Queries the current configuration. The query returns
PER.*/
```

### 3.10.9 CONFigure:RESistance

**Syntax**

CONFigure:RESistance [{<range>|<lim>}[, {<resolution>|<lim>}]]

**Description**

Presets the multimeter with the specified range and resolution for 2-wire resistance measurement.

**Parameter**

Name	Type	Range	Default
<range>	Discrete	{100Ω 1kΩ 10kΩ 100kΩ 1MΩ 10MΩ 50MΩ AUTO}	AUTO
<lim>	Discrete	{MIN MAX DEF}	-
<resolution>	Real	Refer to <i>Table 3.14 Resolution, Measurement Speed, and Integration Time</i>	10 ppm x range

**Remarks**

- The default unit of <range> and <resolution> is Ω. You can set the parameters in scientific notation, for example, 1E-3 (0.001 Ω). If the parameter is omitted, it is interpreted as the default value.
- AUTO (auto range) automatically selects the range according to the input signal.



- This command only specifies the parameters without initiating the measurement.

### Return Format

None.

### Examples

```
CONFigure:RESistance 1000,1E-2 /*Sets the range to 1 kΩ and the
resolution to 10 mΩ for 2-wire resistance measurement.*/
CONFigure? /*Queries the current configuration. The query returns
RES 1.000000000E+03,1.000000000E-02.*/
```

## 3.10.10 CONFigure:TEMPerature

### Syntax

**CONFigure:TEMPerature** [{<probe\_type>|<lim>}[,<type>]]

### Description

Resets all measurement parameters and trigger parameters to their default values for temperature measurement. You can specify the type of temperature sensor and sensor parameters.

### Parameter

Name	Type	Range	Default
<probe_type>	Discrete	{FRTD RTD FTHermistor THERmistor TCouple}	TCouple
<lim>	Discrete	{DEFault}	-
<type>	Discrete	{385 389 391 392 2200 3000 5000 10000 30000 B E J K N R S T}	-

### Remarks

- <probe\_type> is used to set the temperature sensor type.
  - **FRTD**: 4-wire thermal resistance
  - **RTD**: 2-wire thermal resistance
  - **FTHermistor**: 4-wire thermistor
  - **THERmistor**: 2-wire thermistor
  - **TCouple**: thermocouple
- <type> is used to set the parameters for temperature sensors.

- When the temperature sensor type is set to 4-wire thermal resistance (FRTD) or 2-wire thermal resistance (RTD), <type> can set the temperature coefficient to 385, 389, 391, or 392.
  - When the temperature sensor type is set to 4-wire thermistor (FTHermistor) or 2-wire thermistor (THERmistor), <type> can set the sensor resistance to 2200 (2.2 k $\Omega$ ), 3000 (3 k $\Omega$ ), 5000 (5 k $\Omega$ ), 10000 (10 k $\Omega$ ), or 30000 (30 k $\Omega$ ).
  - When the temperature sensor type is set to thermocouple (TCouple), <type> can set the sensor type to B (Pt Rh 30-Pt Rh6), E (NI CR-WRCK), J (Fe-WRCK), K (NI CR-NiSi), N (NiCrNi-NiSi), R (Pt Rh13 -Pt), S (Pt Rh10-Pt), and T (Cu-WRCK).
- This command only specifies the parameters without initiating the measurement.

### Return Format

None.

### Examples

```
CONFigure:TEMPerature FRTD,385 /*Sets the sensor type to 4-wire
thermal resistance and the sensor resistance to 385.*/
CONFigure? /*Queries the current configuration. The query returns
TEMP FRTD,385.*/
```

## 3.10.11 CONFigure:VOLTage:AC

### Syntax

```
CONFigure:VOLTage:AC [{<range>|<lim>}[,<resolution>|<lim>}]
```

### Description

Presets the multimeter with the specified range and resolution for AC voltage measurement.

### Parameter

Name	Type	Range	Default
<range>	Discrete	{100mV 1V 10V 100V 750V AUTO}	AUTO
<lim>	Discrete	{MIN MAX DEF}	-
<resolution>	Real	Refer to <i>Table 3.14 Resolution, Measurement Speed, and Integration Time</i>	10 ppm x range

**Remarks**

- The default unit of <range> and <resolution> is V. You can set the parameters in scientific notation, for example, 1E-3 (0.001 V). If the parameter is omitted, it is interpreted as the default value.
- AUTO (auto range) automatically selects the range according to the input signal.
- This command only specifies the parameters without initiating the measurement.

**Return Format**

None.

**Examples**

```
CONFigure:VOLTage:AC 10,1E-3 /*Sets the range to 10 V and
resolution to 1 mV for AC voltage measurement.*/
CONFigure? /*Queries the current configuration. The query returns
VOLT:AC 1.00000000E+01,1.00000000E-03.*/
```

### 3.10.12 CONFigure:VOLTage:DC

**Syntax**

```
CONFigure:VOLTage:DC [{<range>|<lim>}[, {<resolution>|<lim>}]]
```

**Description**

Presets the multimeter with the specified range and resolution for DC voltage measurement.

**Parameter**

Name	Type	Range	Default
<range>	Discrete	{100mV 1V 10V 100V 1000V  AUTO}	AUTO
<lim>	Discrete	{MIN MAX DEF}	-
<resolution>	Real	Refer to <i>Table 3.14 Resolution, Measurement Speed, and Integration Time</i>	10 ppm x range

**Remarks**

- The default unit of <range> and <resolution> is V. You can set the parameters in scientific notation, for example, 1E-3 (0.001 V). If the parameter is omitted, it is interpreted as the default value.
- AUTO (auto range) automatically selects the range according to the input signal.
- This command only specifies the parameters without initiating the measurement.

**Return Format**

None.

**Examples**

```
CONFigure:VOLTage:DC 10,1E-3 /*Sets the range to 10 V and
resolution to 1 mV for DC voltage measurement.*/
CONFigure? /*Queries the current configuration. The query returns
VOLT 1.00000000E+01,1.00000000E-03.*/
```

### 3.10.13 CONFigure?

**Syntax**

CONFigure?

**Description**

Queries the current configuration.

**Parameter**

None.

**Remarks**

None.

**Return Format**

The query returns a string, indicating the current measurement function, range, and resolution (if any). For example, the query might return VOLT:AC 1.00000000E+01,1.00000000E-03, indicating that the current measurement function is AC voltage measurement, the range is 10 V, and the resolution is 1 mV.

**Examples**

```
CONFigure? /*Queries the current configuration. The query returns
VOLT:AC 1.00000000E+01,1.00000000E-03.*/
```

## 3.11 DATA Commands

**DATA** commands are used to set or remove data from the reading memory.

### 3.11.1 DATA:LAST?

**Syntax**

DATA:LAST?

**Description**

Queries the last performed measurement data and measurement function.

**Parameter**

None.

**Remarks**

None.

**Return Format**

The query returns a string in the format of measurement data, measurement function. For example, the query might return -5.07000000E-01 VDC, indicating that the last performed measurement data is -5.07000000E-01 and the measurement function is DC voltage measurement. If there is no available measurement data, the query returns 9.90000000E+37.

**Examples**

```
DATA:LAST? /*Queries the last performed measurement data. The query
returns -5.07000000E-01 VDC.*/
```

### 3.11.2 DATA:POINTs?

**Syntax**

**DATA:POINTs?**

**Description**

Queries the total number of readings currently stored in reading memory.

**Parameter**

None.

**Remarks**

DM858 can store up to 500,000 readings while DM858E can store up to 20,000 readings.

**Return Format**

The query returns an integer.

**Examples**

```
DATA:POINTs? /*Queries the total number of readings currently
stored in reading memory. The query might return 1099.*/
```

### 3.11.3 DATA:POINTs:EVENT:THReshold

**Syntax**

**DATA:POINTs:EVENT:THReshold <count>**

**DATA:POINTs:EVENT:THReshold?**

**Description**

Sets or queries the memory threshold.

**Parameter**

Name	Type	Range	Default
<count>	Integer	DM858: 1 to 500,000	DM858: 500,000
		DM858E: 1 to 20,000	DM858E: 20,000

**Remarks**

The total number of readings stored in the memory cannot exceed the threshold specified by this command.

**Return Format**

The query returns an integer.

**Examples**

```
DATA:POINTs:EVENT:THReshold 10 /*Sets the memory threshold to 10.*/
DATA:POINTs:EVENT:THReshold? /*Queries the memory threshold. The
query returns 10.*/
```

**3.11.4 DATA:REMOve?****Syntax**

**DATA:REMOve?** <num>[,<wait>]

**Description**

Reads and removes the specified number of measurement results from reading memory.

**Parameter**

Name	Type	Range	Default
<num>	Integer	DM858: 1 to 500,000	-
		DM858E: 1 to 20,000	
<wait>	Discrete	{WAIT}	-

**Remarks**

- If <num> is greater than available measurements, the query returns an error; when <wait> is specified and <num> is greater than available measurements, the query waits until <num> is available.

- The *R?* and *DATA:REMove?* queries can be used during a long series of readings to periodically remove readings from the reading memory that would cause the memory to overflow.
- You can store up to 500,000 measurements (DM858)/20,000 measurements (DM858E). If reading memory overflows, new measurements overwrite the oldest measurements stored; the most recent measurements are always preserved.

#### Return Format

The query returns a string.

#### Examples

```
DATA:REMove? 3 /*Reads and removes the three oldest readings from
reading memory. The query returns 1.21770000E+02;9.85760000E
+02;9.86260000E+02.*/
```

## 3.12 HCOPY Commands

**HCOPY** commands are used to set or query the image format, and capture the screenshot.

### 3.12.1 HCOPY:SDUMp:DATA?

#### Syntax

```
HCOPY : SDUMp : DATA?
```

#### Description

Returns the front-panel display (screen shot).

#### Parameter

None.

#### Remarks

The image format is specified by *HCOPY:SDUMp:DATA:FORMat*.

#### Return Format

The query returns a definite-length binary block containing the image.

#### Examples

```
HCOPY:SDUMp:DATA? /*Captures and returns the front-panel display
image.*/
```

### 3.12.2 HCOPY:SDUMp:DATA:FORMat

#### Syntax

HCOPY:SDUMp:DATA:FORMat <type>

HCOPY:SDUMp:DATA:FORMat?

#### Description

Sets or queries the image format.

#### Parameter

Name	Type	Range	Default
<type>	Discrete	{PNG BMP}	PNG

#### Remarks

None.

#### Return Format

The query returns PNG or BMP.

#### Examples

```
HCOPY:SDUMp:DATA:FORMat BMP /*Sets the image format to BMP.*/
HCOPY:SDUMp:DATA:FORMat? /*Queries the image format. The query
returns BMP.*/
```

## 3.13 IEEE488.2 Common Commands

The IEEE488.2 common commands are used to query the basic information of the instrument or executing basic operations. These commands usually start with "\*", and the command keywords contain 3 characters and are related with status registers.

The standard event status register (SESR) and status byte register (SBR) record the event of a certain type happened during the use of the instrument. IEEE488.2 defines to record one specific type of event for each bit in the status register.

**Table 3.26 Table of the Bit Definition of Standard Event Status Register**

Bit No.	Bit Name	Decimal Value	Description
0	Operation Complete (OPC)	1	"Operation complete" indicates that all pending operations were completed following the execution of the command.
1	Not Used	2	-



Bit No.	Bit Name	Decimal Value	Description
2	Query Error (QYE)	4	The instrument tries to read the output buffer but it was empty. Or, a new command line was received before a previous query has been read. Or, both the input buffer and output buffer are full.
3	Device-Specific Error (DDE)	8	Indicates that an error has occurred that is neither a Command Error, a Query Error, nor an Execution Error. A Device-Specific Error is any executed device operation that did not properly complete due to some condition, such as self-check error, calibration error, or other device-specific errors.
4	Execution Error (E)	16	An execution error occurred.
5	Command Error (CME)	32	A command error (command syntax error) has occurred.
6	Not Used	64	-
7	Power On (PON)	128	Indicates that an off-to-on transition has occurred in the device's power supply since last reading or the event register was cleared.

**Table 3.27 Table of the Bit Definition of Status Byte Register**

Bit No.	Bit Name	Decimal Value	Description
0	Not Used	1	-
1	Not Used	2	-
2	Error Queue	4	1 or multiple errors in the error queue
3	Questionable Data Summary	8	Sets 1 or multiple bits (must be the enabled bit) in the questionable data register.

Bit No.	Bit Name	Decimal Value	Description
4	Message Available (MAV)	16	Indicates the available data in the output buffer.
5	Standard Event Summary	32	Sets 1 or multiple bits (must be the enabled bit) in the standard event register.
6	Master Summary Status (MSS)	64	Sets 1 or multiple bits (must be the enabled bit) in the Status Byte Register and generate the service request.
7	Operation Status Register	128	Sets 1 or multiple bits (must be the enabled bit) in the Operation Status Register.

### 3.13.1 \*CLS

#### Syntax

\*CLS

#### Description

Clears all the event registers, and also clears the error queue.

#### Parameter

None.

#### Remarks

None.

#### Return Format

None.

#### Example

None.

### 3.13.2 \*ESE

#### Syntax

\*ESE <maskargument>

\*ESE?

**Description**

Sets or queries the enable register of the standard event register set.

**Parameter**

Name	Type	Range	Default
<maskargument>	Integer	0 to 255	0

**Remarks**

For the definitions of the bits in the standard event register, refer to [Table 3.26 Table of the Bit Definition of Standard Event Status Register](#). The value of <maskargument> is the sum of the decimal values of all bits set in the standard event register. For example, to enable Bit 2 (4 in decimal), Bit 3 (8 in decimal), and Bit 7 (128 in decimal), set the <maskargument> to 140 (4+8+128).

**Return Format**

The query returns an integer. The integer equals to the decimal-weighted sum of all the bits set in the register.

**Example**

```
*ESE 16 /*Enables Bit 4 (16 in decimal) in the register.*/
*ESE? /*The query returns the enable value of the register 16.*/
```

**3.13.3 \*ESR?****Syntax**

**\*ESR?**

**Description**

Queries and clears the event register of the standard event status register.

**Parameter**

None.

**Remarks**

Bit 1 and Bit 6 in the standard event status register ([Table 3.26 Table of the Bit Definition of Standard Event Status Register](#)) are not used and are always treated as 0; therefore, the range of the returned value is a decimal number corresponding to a binary number X0XXXX0X (X is 1 or 0).

**Return Format**

The query returns an integer. The integer equals to the binary-weighted sum of all the bits set in the register.

**Example**

None.

### 3.13.4 \*IDN?

**Syntax**

\*IDN?

**Description**

Queries the ID string of the instrument.

**Parameter**

None.

**Remarks**

None.

**Return Format**

The query returns RIGOL TECHNOLOGIES,<model>,<serial number>,<software version>.

- **<model>**: indicates the model number of the instrument.
- **<serial number>**: indicates the serial number of the instrument.
- **<software version>**: indicates the software version of the instrument.

**Example**

None.

### 3.13.5 \*OPC

**Syntax**

\*OPC

\*OPC?

**Description**

The \*OPC command sets bit 0 (Operation Complete, OPC) in the standard event register to 1 after the current operation is finished.

The \*OPC? command queries whether the current operation is finished.

**Parameter**

N/A

**Remarks**

For the definitions of the bits in the standard event register, refer to [Table 3.26 Table of the Bit Definition of Standard Event Status Register](#).

**Return Format**

The query returns 1 after the current operation is finished; otherwise, the query returns 0.

**Example**

N/A

**3.13.6 \*PSC****Syntax**

`*PSC <bool>`

`*PSC?`

**Description**

Sets or queries the on/off status of clearing the enable bit of the register at power on.

**Parameter**

Name	Type	Range	Default
<bool>	Bool	{0 1}	0

**Remarks**

- \*PSC 1 indicates clearing the enable bit of the register at power on; \*PSC 0 indicates not affecting the enable bit of the register at power on.
- You can also send *\*SRE* (\*SRE 0) or *\*ESE* (\*ESE 0) to clear the enable register of the Status Byte register and Standard Event register.

**Return Format**

The query returns 0 or 1.

**Examples**

```
*PSC 1 /*Enables the power-on clearing of the register enable bit.*/
*PSC? /*Queries the on/off status of clearing the enable bit of the
register at power on. The query returns 1.*/
```

**3.13.7 \*RST****Syntax**

`*RST`

**Description**

Restores the instrument to its factory default settings.

**Parameter**

N/A

**Remarks**

N/A

**Return Format**

N/A

**Example**

N/A

**3.13.8 \*SRE****Syntax**

**\*SRE** <maskargument>

**\*SRE?**

**Description**

Sets or queries the enable register of the status byte register set.

**Parameter**

Name	Type	Range	Default
<maskargument>	Integer	0 to 255	0

**Remarks**

For the definitions of the bits in the status byte register, refer to [Table 3.27 Table of the Bit Definition of Status Byte Register](#). The value of <maskargument> is the sum of the decimal values of all bits set in the status byte register. For example, to enable Bit 2 (4 in decimal), Bit 3 (8 in decimal), and Bit 7 (128 in decimal), set the <maskargument> to 140 (4+8+128).

**Return Format**

The query returns an integer. The integer equals to the decimal-weighted sum of all the bits set in the register.

**Example**

```
*SRE 16 /*Enables Bit 4 (16 in decimal) in the register.*/
*SRE? /*The query returns the enable value of the register 16.*/
```

### 3.13.9 \*STB?

---

**Syntax**

\*STB?

**Description**

Queries the event register for the status byte register. After executing the command, the value in the status byte register is cleared.

**Parameter**

None.

**Remarks**

Bit 0 and Bit 1 in the status byte register (*Table 3.27 Table of the Bit Definition of Status Byte Register*) are not used and are always treated as 0; therefore, the range of the returned value is a decimal number corresponding to a binary number XXXXXX00 (X is 1 or 0).

**Return Format**

The query returns an integer. The integer equals to the decimal-weighted sum of all the bits set in the register.

**Example**

None.

### 3.13.10 \*TRG

---

**Syntax**

\*TRG

**Description**

Generates a trigger event.

**Parameter**

None.

**Remarks**

It is valid only when the trigger source is set to BUS (*TRIGger:SOURce*) and the instrument is in the wait-for-trigger state.

**Return Format**

None.

**Examples**

```
*TRG /*Generates a trigger event.*/*
```

## 3.14 LXI Commands

LXI commands are used to set parameters for LAN and mDNS.

### 3.14.1 LXI:MDNS:ENABLE

**Syntax**

```
LXI:MDNS:ENABLE <bool>
```

```
LXI:MDNS:ENABLE?
```

**Description**

Sets or queries the on/off status of the multicast Domain Name System (mDNS).

**Parameter**

Name	Type	Range	Default
<bool>	Bool	{0 1 OFF ON}	0 OFF

**Remarks**

Enabling the mDNS provides the capability of a DNS server in a small network without a DNS server.

**Return Format**

The query returns 0 or 1.

**Examples**

```
LXI:MDNS:ENABLE ON /*Enables the mDNS.*/*
LXI:MDNS:ENABLE? /*Queries whether the mDNS is enabled. The query
returns 1.*/*
```

### 3.14.2 LXI:MDNS:HNAME[:RESolved]?

**Syntax**

```
LXI:MDNS:HNAME[:RESolved]?
```

**Description**

Queries the resolved mDNS hostname.

**Parameter**

None.



**Remarks**

None.

**Return Format**

The query returns a string enclosed by double quotation marks.

**Examples**

```
None.
```

### 3.14.3 LXI:MDNS:SNAME:DESired

**Syntax**

```
LXI:MDNS:SNAME:DESired "<name>"
```

```
LXI:MDNS:SNAME:DESired?
```

**Description**

Sets or queries the mDNS service name.

**Parameter**

Name	Type	Range	Default
<name>	ASCII string	Refer to <i>Remarks</i>	-

**Remarks**

The parameter specified by <name> should be enclosed by double quotation marks. The maximum length is 30 characters.

**Return Format**

The query returns a string enclosed by double quotation marks.

**Examples**

```
LXI:MDNS:SNAME:DESired "rigoldM" /*Sets the mDNS service name to
rigoldM.*/
LXI:MDNS:SNAME:DESired? /*Queries the mDNS service name. The query
returns "rigoldM".*/
```

### 3.14.4 LXI:MDNS:SNAME[:RESolved]?

**Syntax**

```
LXI:MDNS:SNAME[:RESolved]?
```

**Description**

Queries the resolved mDNS service name.

**Parameter**

None.

**Remarks**

None.

**Return Format**

The query returns a string enclosed by double quotation marks.

**Examples**

```
LXI:MDNS:SNAME:RESolved? /*Queries the resolved mDNS service name.  
The query might return "RIGOL MULTIMETER".*/
```

### 3.14.5 LXI:RESet

---

**Syntax**

```
LXI:RESet
```

**Description**

Resets the LAN parameters to default values.

**Parameter**

None.

**Remarks**

None.

**Return Format**

None.

**Examples**

```
LXI:RESet /*Resets the LAN parameters to default values.*/
```

### 3.14.6 LXI:REStArt

---

**Syntax**

```
LXI:REStArt
```

**Description**

Applies the current LAN settings and restart LAN.

**Parameter**

None.

**Remarks**

Depending on your network, the LAN interface may take several seconds to restart after this command is sent.

**Return Format**

None.

**Examples**

```
LXI:REStart /*Applies the current LAN settings and restart LAN.*/
```

## 3.15 MEASure Commands

**MEASure** commands can preset the multimeter with the specified range and resolution, perform the specified measurement function, and then send the reading to the output buffer. The query returns the data in the output buffer.

**TIP**

When using **MEASure** for measurements, you can select the function, range, and resolution. The other measurement parameters are preset to their default values.

### 3.15.1 MEASure:CAPacitance?

**Syntax**

```
MEASure:CAPacitance? [{<range>|<lim>}]
```

**Description**

Presets the multimeter with the specified capacitance range, performs the capacitance measurement, and then sends the reading to the output buffer.

**Parameter**

Name	Type	Range	Default
<range>	Discrete	{1nF 10nF 100nF 1μF 10μF 100μF 1mF 10mF AUTO}	AUTO
<lim>	Discrete	{MIN MAX DEF}	-

**Remarks**

- Only DM858 supports the 10 mF range. For DM858E, the maximum capacitance range is 1 mF.
- The default unit of <range> is F. You can set the value in scientific notation, for example, 1E-6 (1 μF). If the parameter is omitted, it is interpreted as the default value.

- AUTO (auto range) automatically selects the range according to the input signal.

#### Return Format

The query returns the reading in scientific notation. The query returns 9.9E37 when the measurement value exceeds the measurement range.

#### Examples

```
MEASure:CAPacitance? 1E-8 /*Presets the multimeter with 10 nF range
and performs the capacitance measurement. The query returns
+3.0153000E-09.*/
```

### 3.15.2 MEASure:CONTInuity?

#### Syntax

```
MEASure:CONTInuity?
```

#### Description

Sets the continuity measurement parameters and trigger parameters to their default values, performs the continuity measurement, and then sends the reading to the output buffer.

#### Parameter

None.

#### Remarks

None.

#### Return Format

The query returns the measurement value. If the measurement value exceeds 1.2 k $\Omega$ , the query returns 9.9E37.

#### Examples

```
MEASure:CONTInuity? /*Configures the multimeter and performs the
continuity measurement. The query returns 8.40000000E+00.*/
```

### 3.15.3 MEASure:CURRent:AC?

#### Syntax

```
MEASure:CURRent:AC? [{<range>|<lim>}[,<resolution>|<lim>]]
```

#### Description

Presets the multimeter with the specified range and resolution, performs AC current measurement, and then sends the reading to the output buffer.

**Parameter**

Name	Type	Range	Default
<range>	Discrete	{100μA 1mA 10mA 100mA 1A 3A 10A AUTO}	AUTO
<lim>	Discrete	{MIN MAX DEF}	-
<resolution>	Real	Refer to <i>Table 3.14 Resolution, Measurement Speed, and Integration Time</i>	10 ppm x range

**Remarks**

- For <range>, 3 A range is only available for DM858E while 10 A range is only available for DM858.
- The default unit of <range> and <resolution> is A. You can set the parameters in scientific notation, for example, 1E-3 (0.001 A). If the parameter is omitted, it is interpreted as the default value.
- AUTO (auto range) automatically selects the range according to the input signal.

**Return Format**

The query returns the AC current measurement value in scientific notation. For example, the query might return 3.19800000E-02.

**Examples**

```
MEASure:CURRent:AC? 1,1E-5 /*Presets the multimeter with 1 A range and 10 μA resolution and performs the AC current measurement. The query returns 3.19800000E-02.*/
```

### 3.15.4 MEASure:CURRent:DC?

**Syntax**

```
MEASure:CURRent:DC? [{<range>|<lim>}[, {<resolution>|<lim>}]]
```

**Description**

Presets the multimeter with the specified range and resolution, performs DC current measurement, and then sends the reading to the output buffer.

**Parameter**

Name	Type	Range	Default
<range>	Discrete	{100μA 1mA 10mA 100mA 1A 3A 10A AUTO}	AUTO

Name	Type	Range	Default
<lim>	Discrete	{MIN MAX DEF}	-
<resolution>	Real	Refer to <i>Table 3.14 Resolution, Measurement Speed, and Integration Time</i>	10 ppm x range

**Remarks**

- For <range>, 3 A range is only available for DM858E while 10 A range is only available for DM858.
- The default unit of <range> and <resolution> is A. You can set the parameters in scientific notation, for example, 1E-3 (0.001 A). If the parameter is omitted, it is interpreted as the default value.
- AUTO (auto range) automatically selects the range according to the input signal.

**Return Format**

The query returns the DC current measurement value in scientific notation. For example, the query might return 1.20000000E-01.

**Examples**

```
MEASure:CURRent:DC? 1,1E-5 /*Presets the multimeter with 1 A range and 10 µA resolution and performs the DC current measurement. The query returns 1.20000000E-01.*/
```

### 3.15.5 MEASure:DIODE?

**Syntax**

```
MEASure:DIODE?
```

**Description**

Sets the diode measurement parameters and trigger parameters to their default values, performs the diode measurement, and then sends the reading to the output buffer.

**Parameter**

None.

**Remarks**

The range of the diode measurement is fixed to 2 V.

**Return Format**

The measurement value is returned if the measuring voltage is less than 2.1 V. If the measuring voltage is greater than 2.1 V or there is an open circuit, the query returns 9.9E37.

**Examples**

```
MEASure:DIODE? /*Configures the multimeter and performs the diode measurement. The query returns 1.40000000E-01.*/
```

**3.15.6 MEASure:FREQuency?****Syntax**

```
MEASure:FREQuency?
```

**Description**

Sets the frequency measurement parameters and trigger parameters to their default values, performs the frequency measurement, and then sends the reading to the output buffer.

**Parameter**

None.

**Remarks**

None.

**Return Format**

The query returns the frequency value in scientific notation. The query returns 0 if no measuring signal is connected.

**Examples**

```
MEASure:FREQuency? /*Configures the multimeter and performs the frequency measurement. The query returns 2.40000000E+02.*/
```

**3.15.7 MEASure:FRESistance?****Syntax**

```
MEASure:FRESistance? [{<range>|<lim>}[, {<resolution>|<lim>}]]
```

**Description**

Presets the multimeter with the specified range and resolution, performs the 4-wire resistance measurement, and then sends the reading to the output buffer.

**Parameter**

Name	Type	Range	Default
<range>	Discrete	{100Ω 1kΩ 10kΩ 100kΩ 1MΩ 10MΩ 50MΩ AUTO}	AUTO
<lim>	Discrete	{MIN MAX DEF}	-
<resolution>	Real	Refer to <i>Table 3.14 Resolution, Measurement Speed, and Integration Time</i>	10 ppm x range

**Remarks**

- The default unit of <range> and <resolution> is Ω. You can set the parameters in scientific notation, for example, 1E-3 (0.001 Ω). If the parameter is omitted, it is interpreted as the default value.
- AUTO (auto range) automatically selects the range according to the input signal.

**Return Format**

The query returns the 4-wire resistance measurement value in scientific notation. For example, the query might return 7.50000000E+02.

**Examples**

```
MEASure:FRESistance? 1000,1E-2 /*Presets the multimeter with 1 kΩ range and 10 mΩ resolution and performs the 4-wire resistance measurement. The query returns 7.50000000E+02.*/
```

### 3.15.8 MEASure:PERiod?

**Syntax**

**MEASure:PERiod?**

**Description**

Sets the period measurement parameters and trigger parameters to their default values, performs the period measurement, and then sends the reading to the output buffer.

**Parameter**

None.

**Remarks**

None.



**Return Format**

The query returns the period value in scientific notation. The query returns 9.9E37 if no measuring signal is connected.

**Examples**

```
MEASure:PERiod? /*Configures the multimeter and performs the period measurement. The query returns 2.00000000E-2.*/
```

**3.15.9 MEASure:RESistance?****Syntax**

```
MEASure:RESistance? [{<range>|<lim>}[, {<resolution>|<lim>}]]
```

**Description**

Presets the multimeter with the specified range and resolution, performs the 2-wire resistance measurement, and then sends the reading to the output buffer.

**Parameter**

Name	Type	Range	Default
<range>	Discrete	{100Ω 1kΩ 10kΩ 100kΩ 1MΩ 10MΩ 50MΩ AUTO}	AUTO
<lim>	Discrete	{MIN MAX DEF}	-
<resolution>	Real	Refer to <i>Table 3.14 Resolution, Measurement Speed, and Integration Time</i>	10 ppm x range

**Remarks**

- The default unit of <range> and <resolution> is Ω. You can set the parameters in scientific notation, for example, 1E-3 (0.001 Ω). If the parameter is omitted, it is interpreted as the default value.
- AUTO (auto range) automatically selects the range according to the input signal.

**Return Format**

The query returns the 2-wire resistance measurement value in scientific notation. For example, the query might return 7.50000000E+02.

**Examples**

```
MEASure:RESistance? 1000,1E-2 /*Presets the multimeter with 1 kΩ range and 10 mΩ resolution and performs the 2-wire resistance measurement. The query returns 7.50000000E+02.*/
```

### 3.15.10 MEASure:TEMPerature?

#### Syntax

**MEASure:TEMPerature?** [{<probe\_type>|<lim>}[, <type>]]

#### Description

Resets all measurement parameters and trigger parameters to their default values, performs the temperature measurement, and then sends the reading to the output buffer. You can specify the type of temperature sensor and sensor parameters.

#### Parameter

Name	Type	Range	Default
<probe_type>	Discrete	{FRTD RTD FTHermistor THERmistor TCouple}	TCouple
<lim>	Discrete	{DEfault}	-
<type>	Discrete	{385 389 391 392 2200 3000 5000 10000 30000 B E J K N R S T}	-

#### Remarks

- <probe\_type> is used to set the temperature sensor type.
  - **FRTD**: 4-wire thermal resistance
  - **RTD**: 2-wire thermal resistance
  - **FTHermistor**: 4-wire thermistor
  - **THERmistor**: 2-wire thermistor
  - **TCouple**: thermocouple
- <type> is used to set the parameters for temperature sensors.
  - When the temperature sensor type is set to 4-wire thermal resistance (FRTD) or 2-wire thermal resistance (RTD), <type> can set the temperature coefficient to 385, 389, 391, or 392.
  - When the temperature sensor type is set to 4-wire thermistor (FTHermistor) or 2-wire thermistor (THERmistor), <type> can set the sensor resistance to 2200 (2.2 k $\Omega$ ), 3000 (3 k $\Omega$ ), 5000 (5 k $\Omega$ ), 10000 (10 k $\Omega$ ), or 30000 (30 k $\Omega$ ).
  - When the temperature sensor type is set to thermocouple (TCouple), <type> can set the sensor type to B (Pt Rh 30-Pt Rh6), E (NI CR-WRCK), J (Fe-WRCK), K (NI CR-NiSi), N (NiCrNi-NiSi), R (Pt Rh13 -Pt), S (Pt Rh10-Pt), and T (Cu-WRCK).

### Return Format

The query returns the temperature value in scientific notation.

### Examples

```
MEASure:TEMPerature? FTHERmistor,5000 /*Sets the sensor type to 4-wire thermistor and the sensor resistance to 5 kΩ, and performs the temperature measurement. The query returns 4.87268785E+01.*/
```

## 3.15.11 MEASure:VOLTage:AC?

### Syntax

```
MEASure:VOLTage:AC? [{<range>|<lim>}][,<resolution>|<lim>}]
```

### Description

Presets the multimeter with the specified range and resolution, performs the AC voltage measurement, and then sends the reading to the output buffer.

### Parameter

Name	Type	Range	Default
<range>	Discrete	{100mV 1V 10V 100V 750V AUTO}	AUTO
<lim>	Discrete	{MIN MAX DEF}	-
<resolution>	Real	Refer to <i>Table 3.14 Resolution, Measurement Speed, and Integration Time</i>	10 ppm x range

### Remarks

- The default unit of <range> and <resolution> is V. You can set the parameters in scientific notation, for example, 1E-3 (0.001 V). If the parameter is omitted, it is interpreted as the default value.
- AUTO (auto range) automatically selects the range according to the input signal.

### Return Format

The query returns the AC voltage measurement value in scientific notation. For example, the query might return 7.62810000E+00.

### Examples

```
MEASure:VOLTage:AC? 10,1E-3 /*Presets the multimeter with 10 V range and 1 mV resolution and performs the AC voltage measurement. The query returns 7.62810000E+00.*/
```

### 3.15.12 MEASure:VOLTage:DC?

#### Syntax

```
MEASure:VOLTage:DC? [{<range>|<lim>}][,<resolution>|<lim>}]
```

#### Description

Presets the multimeter with the specified range and resolution, performs DC voltage measurement, and then sends the reading to the output buffer.

#### Parameter

Name	Type	Range	Default
<range>	Discrete	{100mV 1V 10V 100V 1000V AUTO}	AUTO
<lim>	Discrete	{MIN MAX DEF}	-
<resolution>	Real	Refer to <i>Table 3.14 Resolution, Measurement Speed, and Integration Time</i>	10 ppm x range

#### Remarks

- The default unit of <range> and <resolution> is V. You can set the parameters in scientific notation, for example, 1E-3 (0.001 V). If the parameter is omitted, it is interpreted as the default value.
- AUTO (auto range) automatically selects the range according to the input signal.

#### Return Format

The query returns the DC voltage measurement value in scientific notation. For example, the query might return 1.73810000E+00.

#### Examples

```
MEASure:VOLTage:DC? 10,1E-3 /*Presets the multimeter with 10 V range and 1 mV resolution and performs the DC voltage measurement. The query returns 1.73810000E+00.*/
```

## 3.16 MMEMory Commands

**MMEMory** commands are used to manage files in the instrument's internal and external memory, store and load the instrument state. The internal file system ("INT:") of the instrument is always present while the external memory ("USB:") is available only when the front-panel USB HOST interface detects the USB storage device.

### 3.16.1 MMEemory:CATalog[:ALL]?

#### Syntax

**MMEemory:CATalog[:ALL]?** "<folder>"

#### Description

Queries all the files in the specified directory.

#### Parameter

Name	Type	Range	Default
<folder>	ASCII string	Valid directory	-

#### Remarks

The parameter of <folder> should be enclosed by double quotation marks. "<folder>" is the valid directory in the internal or external memory. For example, it can be "INT:\folder" or "USB:\Rigol". If it is omitted, this command queries the directory specified by *MMEemory:CDIRectory*.

#### Return Format

The query returns a string in the format of space used,space available,"file name,file property,file size",.....; wherein, the units of the space used and space available are byte; the file name includes the file extension (if any); the file property is either STAT for STATe (\*.sta) files, ASC for DATA files (\*.csv), PREF for PREFerence files (\*.prf), FOLD for folders, or null for all other file extensions; the file size is expressed in bytes. For example, the query might return +1000000000,+327168572,"command.exe,,375808","MySetup.sta,STAT,8192",MyData.csv,ASC,11265", indicating that the space used is 1000000000 bytes and the space available is 327168572 bytes; it contains one executable file, one state file (MySetup.sta), and one waveform file (MyData.csv).

#### Examples

```
MMEemory:CATalog:ALL? "INT:\Data" /*Queries all the files in the
folder named Data under C disk.*/
```

### 3.16.2 MMEemory:CDIRectory

#### Syntax

**MMEemory:CDIRectory** "<directory\_name>"

**MMEemory:CDIRectory?**

#### Description

Sets or queries the default path for *MMEemory Commands*.

**Parameter**

Name	Type	Range	Default
<directory_name>	ASCII string	Valid directory	INT:\

**Remarks**

The parameter of <directory\_name> should be enclosed by double quotation marks. "<directory\_name>" is the valid directory in the internal or external memory. For example, it can be "INT:\folder" or "USB:\Rigol".

**Return Format**

The query returns a string enclosed by double quotation marks. For example, the query might return "INT:\folder", representing the folder named "folder" under local disk C.

**Examples**

```
MMEMory:CDIRectory "INT:\folder" /*Sets the default directory to
INT:\folder (the "folder" in internal memory).*/
MMEMory:CDIRectory? /*Queries the default directory. The query
returns "INT:\folder".*/
```

### 3.16.3 MMEMory:COPY

**Syntax**

**MMEMory:COPY** "<file1>","<file2>"

**Description**

Copies file 1 to file 2.

**Parameter**

Name	Type	Range	Default
<file1>	ASCII string	Valid filename	-
<file2>	ASCII string	Valid directory or filename	-

**Remarks**

- The parameter of <file1>/<file2> should be enclosed by double quotation marks.
- "<file1>" is the source file. The source file and folder must exist and should include the file extension. For example, it can be set to "INT:\Mysetup.sta".
- If "<file2>" is specified as the filename, then the source file is copied to the specified path with the specified filename. This command overwrites the existing file of the same name under the target path. For example, if "<file2>" is specified

as "INT:\TextFolder\copy.sta", then the source file is copied to the folder named TextFolder in the internal memory with the filename of copy.sta.

- If "<file2>" is specified as a folder, then the source file is copied to the specified path with the same name. For example, if "<file2>" is specified as "INT:\TextFolder\", then the source file is copied to the folder named TextFolder in the internal memory with the same filename.

### Return Format

None.

### Examples

```
MMEemory:COPY "INT:\Mysetup.sta","INT:\TextFolder\" /*Copies the
file named Mysetup.sta under C disk to the directory named
TextFolder under C disk.*/
```

## 3.16.4 MMEemory:DElete

### Syntax

```
MMEemory:DElete "<file_name>"
```

### Description

Deletes a specific file under the specified directory.

### Parameter

Name	Type	Range	Default
<file_name>	ASCII string	Valid filename	-

### Remarks

- The parameter of <file\_name> should be enclosed by double quotation marks. "<file\_name>" is a valid filename in internal or external memory, for example, "INT:\screenshot.png"
- To delete a folder, use *MMEemory:RDIRECTORY*.

### Return Format

None.

### Examples

```
MMEemory:DElete "INT:\screenshot.png" /*Deletes the image named
screenshot.png under C disk.*/
```

### 3.16.5 MMEMory:LOAD:PREFerences

#### Syntax

```
MMEMory:LOAD:PREFerences "<file>"
```

#### Description

Loads nonvolatile I/O settings and user preferences from a specified file.

#### Parameter

Name	Type	Range	Default
<file>	ASCII string	Valid filename	-

#### Remarks

- The parameter of <file> should be enclosed by double quotation marks.
- The preferences are nonvolatile parameters (e.g. system language, beeper on/off, power-on state) related to the instrument.

#### Return Format

None.

#### Examples

```
MMEMory:LOAD:PREFerences "INT:\MyPreferences" /*Loads nonvolatile
I/O settings and user preferences from the file named
"MyPreferences" in C disk.*/
```

### 3.16.6 MMEMory:LOAD:STATE

#### Syntax

```
:MMEMory:LOAD:STATE "<file>"
```

#### Description

Loads the specified state file.

#### Parameter

Name	Type	Range	Default
<file>	ASCII string	Valid name of state file	-

#### Remarks

- The parameter of <file> should be enclosed by double quotation marks. "<file>" is the name of the state file under the specified directory, for example, "INT:\Mystate.sta"



- The specified state file must exist.

### Return Format

None.

### Examples

```
MMEMemory:LOAD:STATE "INT:\Mystate.sta" /*Loads the instrument state
file named MySetup.sta from the root directory of the internal
memory.*/
```

## 3.16.7 MMEMemory:MDIRectory

### Syntax

```
MMEMemory:MDIRectory "<dir_name>"
```

### Description

Creates an empty folder with the specified name in the mass memory system.

### Parameter

Name	Type	Range	Default
<dir_name>	ASCII string	Refer to <i>Remarks</i>	-

### Remarks

- The parameter of <dir\_name> should be enclosed by double quotation marks.
- If the current directory has an existing folder of the same name, a prompt message will be displayed, indicating that errors occur on the remote command.

### Return Format

None.

### Examples

```
MMEMemory:MDIRectory "INT:\TestFolder" /*Creates a folder named
TestFolder in the internal memory.*/
```

## 3.16.8 MMEMemory:MOVE

### Syntax

```
MMEMemory:MOVE "<file1>","<file2>"
```

### Description

Moves file 1 under the current directory to the specified directory, or renames file 1 to file 2.

**Parameter**

Name	Type	Range	Default
<file1>	ASCII string	Valid filename	-
<file2>	ASCII string	Valid directory or filename	-

**Remarks**

- The parameter of <file1>/<file2> should be enclosed by double quotation marks.
- "<file1>" is the file under the specified directory, for example, "INT:\Rigol.sta".
- "<file2>" is the valid directory or filename in the internal or external memory. For example, it can be "INT:\" or "USB:\Rigol\".
- If you want to rename the file, <file1> and <file2> should share the same folder; if you want to move the file, then "<file2>" should be specified as another valid directory such as "USB:\Rigol\".

**Return Format**

None.

**Examples**

```
MMEemory:MOVE "INT:\Rigol.sta","USB:\Rigol\" /*Moves the file named
Rigol.sta in internal memory to the folder named Rigol in the USB
storage device.*/
MMEemory:MOVE "USB:\Rigol1.sta","USB:\Rigol2.sta" /*Names the file
named Rigol1.sta in the USB storage device to Rigol2.sta.*/
```

### 3.16.9 MMEemory:RDIrectory

**Syntax**

**MMEemory:RDIrectory** "<folder>"

**Description**

Deletes the specified directory (empty folder) in the mass memory system.

**Parameter**

Name	Type	Range	Default
<folder>	ASCII string	Folder name of the empty folder	-

**Remarks**

The parameter of <folder> should be enclosed by double quotation marks. You can only delete empty folders, or an error message will be displayed.

**Return Format**

None.

**Examples**

```
MMEMemory:RDIrectory "INT:\TestFolder" /*Deletes the empty folder
named "TestFolder" in the internal mass memory.*/
```

**3.16.10 MMEMemory:STORe:PREFerences****Syntax**

```
MMEMemory:STORe:PREFerences "<file>"
```

**Description**

Stores nonvolatile I/O settings and user preferences to a specified file.

**Parameter**

Name	Type	Range	Default
<file>	ASCII string	Valid filename	-

**Remarks**

The parameter of <file> should be enclosed by double quotation marks. "<file>" is a specified file in internal/external memory. For example, it can be set to "INT:\MyPreferences\set.prf" or "USB:\MyPreferences\set.prf".

**Return Format**

None.

**Examples**

```
MMEMemory:STORe:PREFerences "INT:\MyPreferences\set.prf" /*Stores the
current nonvolatile I/O settings and user preferences to the folder
named "MyPreferences" in C disk and names them to set.prf.*/
```

**3.16.11 MMEMemory:STORe:STATe****Syntax**

```
MMEMemory:STORe:STATe "<file>"
```

**Description**

Stores the current instrument state to a state file (\*.sta).

**Parameter**

Name	Type	Range	Default
<file>	ASCII string	Valid filename	-

**Remarks**

The parameter of <file> should be enclosed by double quotation marks. "<file>" is a specified file in internal/external memory. For example, it can be set to "INT:\setup" or "USB:\setup".

**Return Format**

None.

**Examples**

```
MMEemory:STORe:STATe "INT:\MySetup" /*Stores the current instrument state in the file named "MySetup.sta" under C disk.*/
```

### 3.16.12 MMEemory:STORe:DATA

**Syntax**

```
MMEemory:STORe:DATA RDG_STORE,"<file>"
```

**Description**

Stores all the readings from the reading memory in the specified data file.

**Parameter**

Name	Type	Range	Default
<file>	ASCII string	Valid filename	-

**Remarks**

- The parameter of <file> should be enclosed by double quotation marks.
- If the file extension is specified to .csv or no file extension is specified, the data will be stored in ASCII. If the file extension is specified to .dat, the data will be stored in binary format.
- If the target file already exists, it is overwritten directly.

**Return Format**

None.

**Examples**

```
MMEemory:STORe:DATA RDG_STORE,"INT:\MyVoltMeas.csv" /*Stores all the readings to the file named MyVoltMeas.csv under C disk.*/
```

### 3.16.13 MMEemory:STATe:RECall:AUTO

**Syntax**

```
MMEemory:STATe:RECall:AUTO <bool>
```

**MMEMory:STATe:RECall:AUTO?**

### Description

Sets or queries whether to enable the automatic recall of the last power-on setting when the power is turned on.

### Parameter

Name	Type	Range	Default
<bool>	Bool	{0 1 OFF ON}	0 OFF

### Remarks

- 1|ON: the instrument recalls the instrument state (at last power-off) in memory location when the power is turned on.
- 0|OFF: the instrument uses the factory default values (except the parameters not affected by factory default settings) when the power is turned on.

### Return Format

The query returns 0 or 1.

### Examples

```
MMEMory:STATe:RECall:AUTO ON /*Enables the instrument to recall the
last power-on state when the power is turned on.*/
MMEMory:STATe:RECall:AUTO? /*Queries whether to enable the
automatic recall of the last power-on setting when the power is
turned on. The query returns 1.*/
```

## 3.17 SENSE Commands

**SENSE** commands are used to select the multimeter's measurement function and set the measurement parameters for each measurement function.

### 3.17.1 [SENSE]:CAPacitance:NULL[:STATe]

#### Syntax

```
[SENSE]:CAPacitance:NULL[:STATe] <bool>
```

```
[SENSE]:CAPacitance:NULL[:STATe]?
```

#### Description

Sets or queries the on/off status of the relative operation for capacitance measurements.

**Parameter**

Name	Type	Range	Default
<bool>	Bool	{0 1 OFF ON}	0 OFF

**Remarks**

When the relative operation is enabled, the displayed reading value is the result obtained by subtracting the set relative value from the actual measurement value. That is, reading value = actual measurement value - relative value. You can use `[SENSe]:CAPacitance:NULL:VALue` to set the relative value.

**Return Format**

The query returns 0 or 1.

**Examples**

```
SENSe:CAPacitance:NULL:STATE ON /*Enables the relative operation
for capacitance measurements.*/
SENSe:CAPacitance:NULL:STATE? /*Queries whether the relative
operation for capacitance measurements is enabled. The query
returns 1.*/
```

### 3.17.2 [SENSe]:CAPacitance:NULL:VALue

**Syntax**

`[SENSe]:CAPacitance:NULL:VALue {<value>|<lim>}`

`[SENSe]:CAPacitance:NULL:VALue?`

**Description**

Sets or queries the relative value for capacitance measurements.

**Parameter**

Name	Type	Range	Default
<value>	Real	Refer to <i>Remarks</i>	-
<lim>	Discrete	{MIN MAX DEF}	-

**Remarks**

The range of the relative value is -120% to +120% of the current capacitance measurement range (`[SENSe]:CAPacitance:RANGe`).

**Return Format**

The query returns the relative value in scientific notation.

### Examples

```
SENSe:CAPacitance:NULL:VALue 1E-8 /*Sets the relative value to 10
nF for capacitance measurements.*/
SENSe:CAPacitance:NULL:VALue? /*Queries the relative value for
capacitance measurements. The query returns 1.00000000E-08.*/
```

## 3.17.3 [SENSe]:CAPacitance:NULL:VALue:AUTO

### Syntax

```
[SENSe]:CAPacitance:NULL:VALue:AUTO <bool>
```

```
[SENSe]:CAPacitance:NULL:VALue:AUTO?
```

### Description

Sets or queries the on/off status of the auto relative operation for capacitance measurements.

### Parameter

Name	Type	Range	Default
<bool>	Bool	{0 1 OFF ON}	1 ON

### Remarks

- When the auto relative function is enabled, the relative value is automatically set to the the first capacitance measurement value.
- When `[SENSe]:CAPacitance:NULL:VALue` is used to specify a valid relative value, the auto relative function will be disabled.

### Return Format

The query returns 0 or 1.

### Examples

```
SENSe:CAPacitance:NULL:VALue:AUTO ON /*Enables the auto relative
operation for capacitance measurements.*/
SENSe:CAPacitance:NULL:VALue:AUTO? /*Queries whether the auto
relative operation is enabled for capacitance measurements. The
query returns 1.*/
```

## 3.17.4 [SENSe]:CAPacitance:RANGe

### Syntax

```
[SENSe]:CAPacitance:RANGe {<range>|<lim>}
```

```
[SENSe]:CAPacitance:RANGe?
```

### Description

Sets or queries the range for capacitance measurements.

**Parameter**

Name	Type	Range	Default
<range>	Discrete	{1nF 10nF 100nF 1μF 10μF 100μF 1mF 10mF AUTO}	AUTO
<lim>	Discrete	{MIN MAX DEF}	-

**Remarks**

- Only DM858 supports the 10 mF range. For DM858E, the maximum capacitance range is 1 mF.
- The default unit of <range> is F. You can set the value in scientific notation, for example, 1E-6 (1 μF).
- AUTO (auto range) automatically selects the range according to the input signal. You can use `[SENSe]:CAPacitance:RANGe:AUTO` to set the capacitance measurements to auto range.

**Return Format**

The query returns the range for capacitance measurements in scientific notation.

**Examples**

```
SENSe:CAPacitance:RANGe 1E-3 /*Sets the range for capacitance
measurements to 1 mF.*/
SENSe:CAPacitance:RANGe? /*Queries the range for capacitance
measurements. The query returns 1.00000000E-03.*/
```

### 3.17.5 [SENSe]:CAPacitance:RANGe:AUTO

**Syntax**

`[SENSe]:CAPacitance:RANGe:AUTO <bool>`

`[SENSe]:CAPacitance:RANGe:AUTO?`

**Description**

Sets or queries the on/off status of auto range for capacitance measurements.

**Parameter**

Name	Type	Range	Default
<bool>	Bool	{0 1 OFF ON}	1 ON

**Remarks**

Auto range function automatically selects the range according to the input signal. After you use `[SENSe]:CAPacitance:RANGe` to select a fixed range, the auto range is automatically disabled.



**Return Format**

The query returns 0 or 1.

**Examples**

```
SENSe:CAPacitance:RANGe:AUTO ON /*Enables the auto range for
capacitance measurements.*/
SENSe:CAPacitance:RANGe:AUTO? /*Sets or queries the on/off status
of auto range for capacitance measurements. The query returns 1.*/
```

**3.17.6 [SENSe]:CAPacitance:SECOndary****Syntax**

[SENSe]:CAPacitance:SECOndary <second>

[SENSe]:CAPacitance:SECOndary?

**Description**

Sets or queries the secondary measurement function for capacitance measurement.

**Parameter**

Name	Type	Range	Default
<second>	Discrete	{"OFF" "CALCulate:DATA"}	"OFF"

**Description**

- **"OFF"**: disables the secondary measurement function.
- **"CALCulate:DATA"**: the measurement value prior to any maths operations (including relative operation).

**Return Format**

The query returns "OFF" or "CALC:DATA".

**Examples**

```
SENSe:CAPacitance:SECOndary "CALCulate:DATA" /*Sets the secondary
measurement function to the capacitance measurement without any
Maths operation for capacitance measurement.*/
SENSe:CAPacitance:SECOndary? /*Queries the secondary measurement
function for capacitance measurement. The query returns
"CALC:DATA".*/
```

**3.17.7 [SENSe]:CURRent:AC:NULL[:STATe]****Syntax**

[SENSe]:CURRent:AC:NULL[:STATe] <bool>

[SENSe]:CURRent:AC:NULL[:STATe]?

**Description**

Sets or queries the on/off status of the relative function for AC current measurements.

**Parameter**

Name	Type	Range	Default
<bool>	Bool	{0 1 OFF ON}	0 OFF

**Remarks**

When the relative operation is enabled, the displayed reading value is the result obtained by subtracting the set relative value from the actual measurement value. That is, reading value = actual measurement value - relative value. You can use `[SENSe]:CURRent:AC:NULL:VALue` to set the relative value.

**Return Format**

The query returns 0 or 1.

**Examples**

```
SENSe:CURRent:AC:NULL:STATe ON /*Enables the relative function for
AC current measurements.*/
SENSe:CURRent:AC:NULL:STATe? /*Queries whether the relative
function is enabled for AC current measurements. The query returns
1.*/
```

### 3.17.8 [SENSe]:CURRent:AC:NULL:VALue

**Syntax**

`[SENSe]:CURRent:AC:NULL:VALue {<value>|<lim>}`

`[SENSe]:CURRent:AC:NULL:VALue?`

**Description**

Sets or queries the relative value for AC current measurements.

**Parameter**

Name	Type	Range	Default
<value>	Real	Refer to <i>Remarks</i>	-
<lim>	Discrete	{MIN MAX DEF}	-

**Remarks**

The range of the relative value is -120% to +120% of the current AC current measurement range (`[SENSe]:CURRent:AC:RANGe`).

### Return Format

The query returns the relative value in scientific notation.

### Examples

```
SENSe:CURRent:AC:NULL:VALue 1E-3 /*Sets the relative value to 1 mA
for AC current measurements.*/
SENSe:CURRent:AC:NULL:VALue? /*Queries the relative value for AC
current measurements. The query returns 1.00000000E-03.*/
```

## 3.17.9 [SENSe]:CURRent:AC:NULL:VALue:AUTO

### Syntax

[SENSe]:CURRent:AC:NULL:VALue:AUTO <bool>

[SENSe]:CURRent:AC:NULL:VALue:AUTO?

### Description

Sets or queries the on/off status of the auto relative operation for AC current measurements.

### Parameter

Name	Type	Range	Default
<bool>	Bool	{0 1 OFF ON}	1 ON

### Remarks

- When the auto relative function is enabled, the relative value is automatically set to the the first AC current measurement value.
- When `[SENSe]:CURRent:AC:NULL:VALue` is used to specify a valid relative value, the auto relative function will be disabled.

### Return Format

The query returns 0 or 1.

### Examples

```
SENSe:CURRent:AC:NULL:VALue:AUTO ON /*Enables the auto relative
operation for AC current measurements.*/
SENSe:CURRent:AC:NULL:VALue:AUTO? /*Queries the auto relative
operation for AC current measurements. The query returns 1.*/
```

## 3.17.10 [SENSe]:CURRent:AC:RANGe

### Syntax

[SENSe]:CURRent:AC:RANGe {<range>|<lim>}

[SENSe]:CURRent:AC:RANGe?

**Description**

Sets or queries the range for AC current measurements.

**Parameter**

Name	Type	Range	Default
<range>	Discrete	{100μA 1mA 10mA 100mA 1A 3A 10A AUTO}	AUTO
<lim>	Discrete	{MIN MAX DEF}	-

**Remarks**

- For <range>, 3 A range is only available for DM858E while 10 A range is only available for DM858.
- The default unit of <range> is A. You can set the value in scientific notation, for example, 1E-3 (0.001 A).
- AUTO (auto range) automatically selects the range according to the input signal. You can also use `[SENSe]:CURRent:AC:RANGe:AUTO` to set the auto range for AC current measurements.

**Return Format**

The query returns the range for AC current measurements in scientific notation.

**Examples**

```
SENSe:CURRent:AC:RANGe 1 /*Sets the range to 1 A for AC current
measurements.*/ SENSe:CURRent:AC:RANGe? /*Queries the range for AC
current measurements. The query returns 1.00000000E+00.*/
```

### 3.17.11 [SENSe]:CURRent:AC:RANGe:AUTO

**Syntax**

```
[SENSe]:CURRent:AC:RANGe:AUTO <bool>
```

```
[SENSe]:CURRent:AC:RANGe:AUTO?
```

**Description**

Sets or queries the on/off status of auto range for AC current measurements.

**Parameter**

Name	Type	Range	Default
<bool>	Bool	{0 1 OFF ON}	1 ON

**Remarks**

Auto range function automatically selects the range according to the input signal. After you use `[SENSe]:CURRent:AC:RANGe` to select a fixed range, the auto range is automatically disabled.

**Return Format**

The query returns 0 or 1.

**Examples**

```
SENSe:CURRent:AC:RANGe:AUTO ON /*Enables the auto range for AC
current measurements.*/
SENSe:CURRent:AC:RANGe:AUTO? /*Queries the on/off status of auto
range for AC current measurements. The query returns 1.*/
```

**3.17.12 [SENSe]:CURRent:AC:SECOndary****Syntax**

```
[SENSe]:CURRent:AC:SECOndary <second>
```

```
[SENSe]:CURRent:AC:SECOndary?
```

**Description**

Sets or queries the secondary measurement function for AC current measurement.

**Parameter**

Name	Type	Range	Default
<second>	Discrete	{"OFF" "CALCulate:DATA" "FREQuency" "PERiod"}	"OFF"

**Description**

- **"OFF"**: disables the secondary measurement function.
- **"CALCulate:DATA"**: the measurement value prior to any maths operations (including relative operation).
- **"FREQuency"**: frequency measurement.
- **"PERiod"**: period measurement.

**Return Format**

The query returns "OFF", "CALC:DATA", "FREQ", or "PER".

**Examples**

```
SENSe:CURRent:AC:SECOndary "FREQ" /*Sets the secondary measurement
function to frequency measurement for AC current measurement.*/
SENSe:CURRent:AC:SECOndary? /*Queries the secondary measurement
function for AC current measurement. The query returns "FREQ".*/
```

### 3.17.13 [SENSe]:CURRent:DC:NULL[:STATe]

#### Syntax

```
[SENSe]:CURRent:DC:NULL[:STATe] <bool>
```

```
[SENSe]:CURRent:DC:NULL[:STATe]?
```

#### Description

Sets or queries the on/off status of the relative function for DC current measurements.

#### Parameter

Name	Type	Range	Default
<bool>	Bool	{0 1 OFF ON}	0 OFF

#### Remarks

When the relative operation is enabled, the displayed reading value is the result obtained by subtracting the set relative value from the actual measurement value. That is, reading value = actual measurement value - relative value. You can use [\[SENSe\]:CURRent:DC:NULL:VALue](#) to set the relative value.

#### Return Format

The query returns 0 or 1.

#### Examples

```
SENSe:CURRent:DC:NULL:STATe ON /*Enables the relative function for
DC current measurements.*/
SENSe:CURRent:DC:NULL:STATe? /*Queries whether the relative
function is enabled for DC current measurements. The query returns
1.*/
```

### 3.17.14 [SENSe]:CURRent:DC:NULL:VALue

#### Syntax

```
[SENSe]:CURRent:DC:NULL:VALue {<value>|<lim>}
```

```
[SENSe]:CURRent:DC:NULL:VALue?
```

#### Description

Sets or queries the relative value for DC current measurements.

#### Parameter

Name	Type	Range	Default
<value>	Real	Refer to <a href="#">Remarks</a>	-
<lim>	Discrete	{MIN MAX DEF}	-

**Remarks**

The range of the relative value is -120% to +120% of the current DC current measurement range (*[SENSe]:CURRent:DC:RANGe*).

**Return Format**

The query returns the relative value in scientific notation.

**Examples**

```
SENSe:CURRent:DC:NULL:VALue 1E-3 /*Sets the relative value to 1 mA
for DC current measurements.*/
SENSe:CURRent:DC:NULL:VALue? /*Queries the relative value for DC
current measurements. The query returns 1.00000000E-03.*/
```

**3.17.15 [SENSe]:CURRent:DC:NULL:VALue:AUTO****Syntax**

*[SENSe]:CURRent:DC:NULL:VALue:AUTO <bool>*

*[SENSe]:CURRent:DC:NULL:VALue:AUTO?*

**Description**

Sets or queries the on/off status of the auto relative function for DC current measurements.

**Parameter**

Name	Type	Range	Default
<bool>	Bool	{0 1 OFF ON}	1 ON

**Remarks**

- When the auto relative function is enabled, the relative value is automatically set to the the first DC current measurement value.
- When *[SENSe]:CURRent:DC:NULL:VALue* is used to specify a valid relative value, the auto relative function will be disabled.

**Return Format**

The query returns 0 or 1.

**Examples**

```
SENSe:CURRent:DC:NULL:VALue:AUTO ON /*Enables the auto relative
function for DC current measurements.*/
SENSe:CURRent:DC:NULL:VALue:AUTO? /*Queries whether the auto
relative function is enabled for DC current measurements. The query
returns 1.*/
```

### 3.17.16 [SENSe]:CURRent:DC:RANGe

#### Syntax

```
[SENSe]:CURRent:DC:RANGe {<range>|<lim>}
```

```
[SENSe]:CURRent:DC:RANGe?
```

#### Description

Sets or queries the range for DC current measurements.

#### Parameter

Name	Type	Range	Default
<range>	Discrete	{100μA 1mA 10mA 100mA 1A 3A 10A AUTO}	AUTO
<lim>	Discrete	{MIN MAX DEF}	-

#### Remarks

- For <range>, 3 A range is only available for DM858E while 10 A range is only available for DM858.
- The default unit of <range> is A. You can set the value in scientific notation, for example, 1E-3 (0.001 A).
- AUTO (auto range) automatically selects the range according to the input signal. You can also use `[SENSe]:CURRent:DC:RANGe:AUTO` to set the auto range for DC current measurements.

#### Return Format

The query returns the range for AC current measurements in scientific notation.

#### Examples

```
SENSe:CURRent:DC:RANGe 1 /*Sets the range to 1 A for DC current
measurements.*/ SENSe:CURRent:DC:RANGe? /*Queries the range for DC
current measurements. The query returns 1.00000000E+00.*/
```

### 3.17.17 [SENSe]:CURRent:DC:RANGe:AUTO

#### Syntax

```
[SENSe]:CURRent:DC:RANGe:AUTO <bool>
```

```
[SENSe]:CURRent:DC:RANGe:AUTO?
```

#### Description

Sets or queries the on/off status of auto range for DC current measurements.



**Parameter**

Name	Type	Range	Default
<bool>	Bool	{0 1 OFF ON}	1 ON

**Remarks**

Auto range function automatically selects the range according to the input signal. After you use `[SENSe]:CURRent:DC:RANGe` to select a fixed range, the auto range is automatically disabled.

**Return Format**

The query returns 0 or 1.

**Examples**

```
SENSe:CURRent:DC:RANGe:AUTO ON /*Enables the auto range for DC
current measurements.*/
SENSe:CURRent:DC:RANGe:AUTO? /*Queries the on/off status of auto
range for DC current measurements. The query returns 1.*/
```

### 3.17.18 [SENSe]:CURRent[:DC]:NPLC

**Syntax**

[SENSe]:CURRent[:DC]:NPLC {<plc>|<lim>}

[SENSe]:CURRent[:DC]:NPLC?

**Description**

Sets or queries the integration time in power-line cycles (PLCs) for DC current measurements.

**Parameter**

Name	Type	Range	Default
<plc>	Real	{0.4 5 20}	20
<lim>	Discrete	{MIN MAX DEF}	-

**Remarks**

- Integration time is the period that the instrument's analog-to-digital (A/D) converter samples the input signal for a measurement. A longer integration time provides higher measurement resolution but slower measurement.
- The unit for integration time is PLC. 1PLC = 0.02 s.
- Setting the integration time also affects the measurement resolution. To see the relationship between integration time and resolution, see [Table 3.14 Resolution, Measurement Speed, and Integration Time](#).

**Return Format**

The query returns the multiple of the current PLC in scientific notation.

**Examples**

```
SENSe:CURRent:DC:NPLC 5 /*Sets the integration time to 5PLC for DC
current measurements.*/
SENSe:CURRent:DC:NPLC? /*Queries the integration time for DC
current measurements. The query returns 5.00000000E+00.*/
```

**3.17.19 [SENSe]:CURRent[:DC]:RESolution****Syntax**

[SENSe]:CURRent[:DC]:RESolution {<resolution>|<lim>}

[SENSe]:CURRent[:DC]:RESolution?

**Description**

Sets or queries the resolution for DC current measurements.

**Parameter**

Name	Type	Range	Default
<resolution>	Real	Refer to <i>Table 3.14 Resolution, Measurement Speed, and Integration Time</i>	10 ppm x range
<lim>	Discrete	{MAX MIN DEF}	-

**Remarks**

The default unit of <resolution> is A. You can set the parameters in scientific notation, for example, 1E-4 (0.1 mA).

**Return Format**

The query returns the resolution for DC current measurements in scientific notation.

**Examples**

```
SENSe:CURRent:DC:RESolution 1E-4 /*Sets the resolution to 0.1 mA
for DC current measurements.*/
SENSe:CURRent:DC:RESolution? /*Queries the resolution for DC
current measurements. The query returns 1.00000000E-04.*/
```

**3.17.20 [SENSe]:CURRent[:DC]:SECOndary****Syntax**

[SENSe]:CURRent[:DC]:SECOndary <second>

[SENSe]:CURRent[:DC]:SECOndary?

**Description**

Sets or queries the secondary measurement function for DC current measurement.

**Parameter**

Name	Type	Range	Default
<second>	Discrete	{"OFF" "CALCulate:DATA"}	"OFF"

**Description**

- **"OFF"**: disables the secondary measurement function.
- **"CALCulate:DATA"**: the measurement value prior to any maths operations (including relative operation).

**Return Format**

The query returns "OFF" or "CALC:DATA".

**Examples**

```
SENSe:CURRENT:DC:SECondary "CALCulate:DATA" /*Sets the secondary
measurement function to DC current measurement without any Maths
operation for DC current measurement.*/
SENSe:CURRENT:DC:SECondary? /*Queries the secondary measurement
function for DC current measurement. The query returns
"CALC:DATA".*/
```

### 3.17.21 [SENSe]:DATA2?

**Syntax**

[SENSe]:DATA2?

**Description**

Queries the secondary measurement results.

**Parameter**

None.

**Remarks**

None.

**Return Format**

The query returns the secondary measurement results in scientific notation.

**Examples**

```
CONFigure:CURRENT:DC 1,1E-5 /*Sets the range to 1 A, resolution to
10 µA for DCI measurement.*/
SENSe:CURRENT:DC:SECondary "CALCulate:DATA" /*Sets the secondary
measurement function to DC current measurement without any Maths
```

```
operation for DCI measurement.*/
READ? /*Starts the measurements.*/
SENSe:DATA2? /*Queries the secondary measurement results. The query
returns -1.22110156E-03.*/
```

### 3.17.22 [SENSe]:DATA2:CLEAr[:IMMediate]

#### Syntax

```
[SENSe]:DATA2:CLEAr[:IMMediate]
```

#### Description

Clears the secondary measurement results and disables the secondary measurement function.

#### Parameter

None.

#### Remarks

None.

#### Return Format

None.

#### Examples

```
SENSe:DATA2:CLEAr:IMMediate /*Clears the secondary measurement
results and disables the secondary measurement function.*/
```

### 3.17.23 [SENSe]:FREQuency:NULL[:STATe]

#### Syntax

```
[SENSe]:FREQuency:NULL[:STATe] <bool>
```

```
[SENSe]:FREQuency:NULL[:STATe]?
```

#### Description

Sets or queries the on/off status of the relative function for frequency measurements.

#### Parameter

Name	Type	Range	Default
<bool>	Bool	{0 1 OFF ON}	0 OFF

#### Remarks

When the relative operation is enabled, the displayed reading value is the result obtained by subtracting the set relative value from the actual measurement value.

That is, reading value = actual measurement value - relative value. You can use `[SENSe]:FREQuency:NULL:VALue` to set the relative value.

### Return Format

The query returns 0 or 1.

### Examples

```
SENSe:FREQuency:NULL:STATe ON /*Enables the relative function for
frequency measurements.*/
SENSe:FREQuency:NULL:STATe? /*Queries whether the relative function
is enabled for frequency measurements. The query returns 1.*/
```

## 3.17.24 [SENSe]:FREQuency:NULL:VALue

### Syntax

`[SENSe]:FREQuency:NULL:VALue {<value>|<lim>}`

`[SENSe]:FREQuency:NULL:VALue?`

### Description

Sets or queries the relative value for frequency measurements.

### Parameter

Name	Type	Range	Default
<value>	Real	-120 kHz to 120 kHz	-
<lim>	Discrete	{MIN MAX DEF}	-

### Remarks

None.

### Return Format

The query returns the relative value in scientific notation.

### Examples

```
SENSe:FREQuency:NULL:VALue 1000 /*Sets the relative value to 1 kHz
for frequency measurements.*/
SENSe:FREQuency:NULL:VALue? /*Queries the relative value for
frequency measurements. The query returns 1.00000000E+03.*/
```

## 3.17.25 [SENSe]:FREQuency:NULL:VALue:AUTO

### Syntax

`[SENSe]:FREQuency:NULL:VALue:AUTO <bool>`

`[SENSe]:FREQuency:NULL:VALue:AUTO?`

**Description**

Sets or queries the on/off status of the auto relative function for frequency measurements.

**Parameter**

Name	Type	Range	Default
<bool>	Bool	{0 1 OFF ON}	1 ON

**Remarks**

- When the auto relative function is enabled, the relative value is automatically set to the the first frequency measurement value.
- When `[SENSe]:FREQuency:NULL:VALue` is used to specify a valid relative value, the auto relative function will be disabled.

**Return Format**

The query returns 0 or 1.

**Examples**

```
SENSe:FREQuency:NULL:VALue:AUTO ON /*Enables the auto relative
function for frequency measurements.*/
SENSe:FREQuency:NULL:VALue:AUTO? /*Queries whether the auto
relative function is enabled for frequency measurements. The query
returns 1.*/
```

**3.17.26 [SENSe]:FREQuency:VOLTage:RANGe****Syntax**

```
[SENSe]:FREQuency:VOLTage:RANGe {<range>|<lim>}
```

```
[SENSe]:FREQuency:VOLTage:RANGe?
```

**Description**

Sets or queries the voltage range for frequency measurements.

**Parameter**

Name	Type	Range	Default
<range>	Discrete	{100mV 1V 10V 100V 750V  AUTO}	AUTO
<lim>	Discrete	{MIN MAX DEF}	-

**Remarks**

- The default unit of <range> is V. You can set the value in scientific notation, for example, 1E+2 (100 V).
- AUTO (auto range) automatically selects the range according to the input signal. You can also use `[SENSe]:FREQuency:VOLTage:RANGe:AUTO` to set the frequency measurements to voltage auto range.

**Return Format**

The query returns the voltage range for frequency measurements in scientific notation.

**Examples**

```
SENSe:FREQuency:VOLTage:RANGe 1 /*Sets the voltage range to 1 V for
frequency measurements.*/
SENSe:FREQuency:VOLTage:RANGe? /*Queries the voltage range for
frequency measurements. The query returns 1.00000000E+00.*/
```

**3.17.27 [SENSe]:FREQuency:VOLTage:RANGe:AUTO****Syntax**

```
[SENSe]:FREQuency:VOLTage:RANGe:AUTO <bool>
```

```
[SENSe]:FREQuency:VOLTage:RANGe:AUTO?
```

**Description**

Sets or queries the on/off status of voltage auto range for frequency measurements.

**Parameter**

Name	Type	Range	Default
<bool>	Bool	{0 1 OFF ON}	1 ON

**Remarks**

Auto range function automatically selects the range according to the input signal. After you use `[SENSe]:FREQuency:VOLTage:RANGe` to select a fixed range, the auto range is automatically disabled.

**Return Format**

The query returns 0 or 1.

**Examples**

```
SENSe:FREQuency:VOLTage:RANGe:AUTO ON /*Enables the voltage auto
range function for frequency measurements.*/
SENSe:FREQuency:VOLTage:RANGe:AUTO? /*Queries whether the voltage
auto range function is enabled for frequency measurements. The
query returns 1.*/
```

### 3.17.28 [SENSe]:FREQuency:SEConDary

#### Syntax

```
[SENSe]:FREQuency:SEConDary <second>
```

```
[SENSe]:FREQuency:SEConDary?
```

#### Description

Sets or queries the secondary measurement function for frequency measurement.

#### Parameter

Name	Type	Range	Default
<second>	Discrete	{"OFF" "CALCulate:DATA" "VOLTage:AC"}	"OFF"

#### Description

- **"OFF"**: disables the secondary measurement function.
- **"CALCulate:DATA"**: the measurement value prior to any maths operations (including relative operation).
- **"VOLTage:AC"**: AC voltage measurement.

#### Return Format

The query returns "OFF", "CALC:DATA", or "VOLT:AC".

#### Examples

```
SENSe:FREQuency:SEConDary "VOLTage:AC" /*Sets the secondary
measurement function to AC voltage measurement for frequency
measurement.*/
SENSe:FREQuency:SEConDary? /*Queries the secondary measurement
function for frequency measurement. The query returns "VOLT:AC".*/
```

### 3.17.29 [SENSe]:FREStance:NULL[:STATe]

#### Syntax

```
[SENSe]:FREStance:NULL[:STATe] <bool>
```

```
[SENSe]:FREStance:NULL[:STATe]?
```

#### Description

Sets or queries the on/off status of the relative function for 4-wire resistance measurements.



**Parameter**

Name	Type	Range	Default
<bool>	Bool	{0 1 OFF ON}	0 OFF

**Remarks**

When the relative operation is enabled, the displayed reading value is the result obtained by subtracting the set relative value from the actual measurement value. That is, reading value = actual measurement value - relative value. You can use `[SENSe]:FRESistance:NULL:VALue` to set the relative value.

**Return Format**

The query returns 0 or 1.

**Examples**

```
SENSe:FRESistance:NULL:STATE ON /*Enables the relative function for
4-wire resistance measurements.*/
SENSe:FRESistance:NULL:STATE? /*Queries whether the relative
function is enabled for 4-wire resistance measurements. The query
returns 1.*/
```

### 3.17.30 [SENSe]:FRESistance:NULL:VALue

**Syntax**

```
[SENSe]:FRESistance:NULL:VALue {<value>|<lim>}
```

```
[SENSe]:FRESistance:NULL:VALue?
```

**Description**

Sets or queries the relative value for 4-wire resistance measurements.

**Parameter**

Name	Type	Range	Default
<value>	Real	Refer to <i>Remarks</i>	-
<lim>	Discrete	{MIN MAX DEF}	-

**Remarks**

The range of the relative value is -120% to +120% of the current 4-wire resistance measurement range (`[SENSe]:FRESistance:RANGe`).

**Return Format**

The query returns the relative value in scientific notation.

**Examples**

```
SENSe:FRESistance:NULL:VALue 10 /*Sets the relative value to 10 Ω
for 4-wire resistance measurements.*/
SENSe:FRESistance:NULL:VALue? /*Queries the relative value for 4-
wire resistances measurements. The query returns 1.00000000E+01.*/
```

**3.17.31 [SENSe]:FRESistance:NULL:VALue:AUTO****Syntax**

```
[SENSe]:FRESistance:NULL:VALue:AUTO <bool>
```

```
[SENSe]:FRESistance:NULL:VALue:AUTO?
```

**Description**

Sets or queries the on/off status of the auto relative function for 4-wire resistance measurements.

**Parameter**

Name	Type	Range	Default
<bool>	Bool	{0 1 OFF ON}	1 ON

**Remarks**

- When the auto relative function is enabled, the relative value is automatically set to the the first 4-wire resistance measurement value.
- When `[SENSe]:FRESistance:NULL:VALue` is used to specify a valid relative value, the auto relative function will be disabled.

**Return Format**

The query returns 0 or 1.

**Examples**

```
SENSe:FRESistance:NULL:VALue:AUTO ON /*Enables the auto relative
function for 4-wire resistance measurements.*/
SENSe:FRESistance:NULL:VALue:AUTO? /*Queries whether the auto
relative function is enabled for 4-wire resistance measurements.
The query returns 1.*/
```

**3.17.32 [SENSe]:FRESistance:NPLC****Syntax**

```
[SENSe]:FRESistance:NPLC {<plc>|<lim>}
```

```
[SENSe]:FRESistance:NPLC?
```

**Description**

Sets or queries the integration time in power-line cycles (PLCs) for 4-wire resistance measurements.

**Parameter**

Name	Type	Range	Default
<plc>	Real	{0.4 5 20}	20
<lim>	Discrete	{MIN MAX DEF}	-

**Remarks**

- Integration time is the period that the instrument's analog-to-digital (A/D) converter samples the input signal for a measurement. A longer integration time provides higher measurement resolution but slower measurement.
- The unit for integration time is PLC. 1PLC = 0.02 s.
- Setting the integration time also affects the measurement resolution. To see the relationship between integration time and resolution, see [Table 3.14 Resolution, Measurement Speed, and Integration Time](#).

**Return Format**

The query returns the multiple of the current PLC in scientific notation.

**Examples**

```
SENSe:FRESistance:DC:NPLC 5 /*Sets the integration time to 5PLC for
4-wire resistance measurements.*/
SENSe:FRESistance:DC:NPLC? /*Queries the integration time for 4-
wire resistance measurements. The query returns 5.00000000E+00.*/
```

**3.17.33 [SENSe]:FRESistance:RANGe****Syntax**

```
[SENSe]:FRESistance:RANGe {<range>|<lim>}
```

```
[SENSe]:FRESistance:RANGe?
```

**Description**

Sets or queries the range for 4-wire resistance measurements.

**Parameter**

Name	Type	Range	Default
<range>	Discrete	{100Ω 1kΩ 10kΩ 100kΩ 1MΩ 10MΩ 50MΩ AUTO}	AUTO

Name	Type	Range	Default
<lim>	Discrete	{MIN MAX DEF}	-

**Remarks**

- The default unit of <range> is  $\Omega$ . You can set the value in scientific notation, for example, 1E+3 (1 k $\Omega$ ).
- AUTO (auto range) automatically selects the range according to the input signal.

**Return Format**

The query returns the range for 4-wire resistance measurements in scientific notation.

**Examples**

```
SENSe:FRESistance:RANGe 100 /*Sets the range to 100  $\Omega$  for 4-wire
resistance measurements.*/
SENSe:FRESistance:RANGe? /*Queries the range for 4-wire resistance
measurements. The query returns 1.00000000E+02.*/
```

### 3.17.34 [SENSe]:FRESistance:RANGe:AUTO

**Syntax**

```
[SENSe]:FRESistance:RANGe:AUTO <bool>
```

```
[SENSe]:FRESistance:RANGe:AUTO?
```

**Description**

Sets or queries the on/off status of auto range for 4-wire resistance measurements.

**Parameter**

Name	Type	Range	Default
<bool>	Bool	{0 1 OFF ON}	1 ON

**Remarks**

Auto range function automatically selects the range according to the input signal. After you use `[SENSe]:FRESistance:RANGe` to select a fixed range, the auto range is automatically disabled.

**Return Format**

The query returns 0 or 1.

**Examples**

```
SENSe:FRESistance:RANGe:AUTO ON /*Enables the auto range function
for 4-wire resistance measurements.*/
SENSe:FRESistance:RANGe:AUTO? /*Queries whether the auto range
function is enabled for 4-wire resistance measurements. The query
returns 1.*/
```

### 3.17.35 [SENSe]:FRESistance:RESolution

#### Syntax

```
[SENSe]:FRESistance:RESolution {<resolution>|<lim>}
```

```
[SENSe]:FRESistance:RESolution?
```

#### Description

Sets or queries the resolution for 4-wire resistance measurements.

#### Parameter

Name	Type	Range	Default
<resolution>	Real	Refer to <i>Table 3.14 Resolution, Measurement Speed, and Integration Time</i>	10 ppm x range
<lim>	Discrete	{MAX MIN DEF}	-

#### Remarks

The default unit of <resolution> is  $\Omega$ . You can set the parameters in scientific notation, for example, 1E+2 (100  $\Omega$ ).

#### Return Format

The query returns the resolution for 4-wire resistance measurements in scientific notation.

#### Examples

```
SENSe:FRESistance:RESolution 100 /*Sets the resolution to 100  $\Omega$  for
4-wire resistance measurements.*/
SENSe:FRESistance:RESolution? /*Queries the resolution for 4-wire
resistance measurements. The query returns 1.00000000E+02.*/
```

### 3.17.36 [SENSe]:FRESistance:SECondary

#### Syntax

```
[SENSe]:FRESistance:SECondary <second>
```

```
[SENSe]:FRESistance:SECondary?
```

#### Description

Sets or queries the secondary measurement function for 4-wire resistance measurement.

**Parameter**

Name	Type	Range	Default
<second>	Discrete	{"OFF" "CALCulate:DATA"}	"OFF"

**Description**

- **"OFF"**: disables the secondary measurement function.
- **"CALCulate:DATA"**: the measurement value prior to any maths operations (including relative operation).

**Return Format**

The query returns "OFF" or "CALC:DATA".

**Examples**

```
SENSE:FRESistance:SECondary "CALCulate:DATA" /*Sets the secondary
measurement function to the 4-wire resistance measurement without
any Maths operation for 4-wire resistance measurement.*/
SENSE:FRESistance:SECondary? /*Queries the secondary measurement
function for 4-wire resistance measurement. The query returns
"CALC:DATA".*/
```

### 3.17.37 [SENSe]:FUNctIon

**Syntax**

[SENSe]:FUNctIon <function>

[SENSe]:FUNctIon?

**Description**

Sets or queries the current measurement function.

**Parameter**

Name	Type	Range	Default
<function>	Discrete	Refer to <i>Remarks</i>	-

**Remarks**

The table below shows the parameters and their corresponding measurement functions.

Parameter	Measurement Function	Parameter	Measurement Function
"CAPacitance"	Capacitance	"FRESistance"	4-Wire resistance
"CONTInuity"	Continuity	"PERiod"	Period

Parameter	Measurement Function	Parameter	Measurement Function
"CURRent:AC"	AC current	"RESistance"	2-Wire resistance
"CURRent[:DC]"	DC current	"TEMPerature"	Temperature
"DIODE"	Diode	"VOLTage:AC"	AC voltage
"FREQuency"	Frequency	"VOLTage[:DC]"	DC voltage

### Return Format

The query returns "CAP", "CONT", "CURR:AC", "CURR", "DIOD", "FREQ", "FRES", "PER", "RES", "TEMP", "VOLT:AC", or "VOLT".

### Examples

```
SENSe:FUNction "CURRent:AC" /*Sets the current measurement function
to AC current measurement.*/
SENSe:FUNction? /*Queries the current measurement function. The
query returns "CURR:AC".*/
```

## 3.17.38 [SENSe]:PERiod:NULL[:STATe]

### Syntax

[SENSe]:PERiod:NULL[:STATe] <bool>

[SENSe]:PERiod:NULL[:STATe]?

### Description

Sets or queries the on/off status of the relative function for period measurements.

### Parameter

Name	Type	Range	Default
<bool>	Bool	{0 1 OFF ON}	0 OFF

### Remarks

When the relative operation is enabled, the displayed reading value is the result obtained by subtracting the set relative value from the actual measurement value. That is, reading value = actual measurement value - relative value. You can use [\[SENSe\]:PERiod:NULL:VALue](#) to set the relative value.

### Return Format

The query returns 0 or 1.

### Examples

```
SENSe:PERiod:NULL:STATe ON /*Enables the relative function for
period measurements.*/
```

```
SENSe:PERiod:NULL:STATe? /*Queries whether the relative function is
enabled for period measurements. The query returns 1.*/
```

### 3.17.39 [SENSe]:PERiod:NULL:VALue

#### Syntax

```
[SENSe]:PERiod:NULL:VALue {<value>|<lim>}
```

```
[SENSe]:PERiod:NULL:VALue?
```

#### Description

Sets or queries the relative value for period measurements.

#### Parameter

Name	Type	Range	Default
<value>	Real	-60 ms to +60 ms	-
<lim>	Discrete	{MIN MAX DEF}	-

#### Remarks

None.

#### Return Format

The query returns the relative value in scientific notation.

#### Examples

```
SENSe:PERiod:NULL:VALue 1E-3 /*Sets the relative value to 1 ms for
period measurements.*/
SENSe:PERiod:NULL:VALue? /*Queries the relative value for period
measurements. The query returns 1.00000000E-03.*/
```

### 3.17.40 [SENSe]:PERiod:NULL:VALue:AUTO

#### Syntax

```
[SENSe]:PERiod:NULL:VALue:AUTO <bool>
```

```
[SENSe]:PERiod:NULL:VALue:AUTO?
```

#### Description

Sets or queries the on/off status of the auto relative function for period measurements.

#### Parameter

Name	Type	Range	Default
<bool>	Bool	{0 1 OFF ON}	1 ON



**Remarks**

- When the auto relative function is enabled, the relative value is automatically set to the the first period measurement value.
- When `[SENSe]:PERiod:NULL:VALue` is used to specify a valid relative value, the auto relative function will be disabled.

**Return Format**

The query returns 0 or 1.

**Examples**

```
SENSe:PERiod:NULL:VALue:AUTO ON /*Enables the auto relative
function for period measurements.*/
SENSe:PERiod:NULL:VALue:AUTO? /*Queries whether the auto relative
function is enabled for period measurements. The query returns 1.*/
```

**3.17.41 [SENSe]:PERiod:VOLTage:RANGe****Syntax**

```
[SENSe]:PERiod:VOLTage:RANGe {<range>|<lim>}
```

```
[SENSe]:PERiod:VOLTage:RANGe?
```

**Description**

Sets or queries the voltage range for period measurements.

**Parameter**

Name	Type	Range	Default
<range>	Discrete	{100mV 1V 10V 100V 750V  AUTO}	AUTO
<lim>	Discrete	{MIN MAX DEF}	-

**Remarks**

- The default unit of <range> is V. You can set the value in scientific notation, for example, 1E+2 (100 V).
- AUTO (auto range) automatically selects the range according to the input signal. You can also use `[SENSe]:PERiod:VOLTage:RANGe:AUTO` to set the period measurements to voltage auto range.

**Return Format**

The query returns the voltage range for period measurements in scientific notation.

**Examples**

```
SENSe:PERiod:VOLTage:RANGe 1 /*Sets the voltage range to 1 V for
period measurements.*/
SENSe:PERiod:VOLTage:RANGe? /*Queries the voltage range for period
measurements. The query returns 1.00000000E+00.*/
```

**3.17.42 [SENSe]:PERiod:VOLTage:RANGe:AUTO****Syntax**

```
[SENSe]:PERiod:VOLTage:RANGe:AUTO <bool>
```

```
[SENSe]:PERiod:VOLTage:RANGe:AUTO?
```

**Description**

Sets or queries the on/off status of voltage auto range for period measurements.

**Parameter**

Name	Type	Range	Default
<bool>	Bool	{0 1 OFF ON}	1 ON

**Remarks**

Auto range function automatically selects the range according to the input signal. After you use `[SENSe]:PERiod:VOLTage:RANGe` to select a fixed range, the auto range is automatically disabled.

**Return Format**

The query returns 0 or 1.

**Examples**

```
SENSe:PERiod:VOLTage:RANGe:AUTO ON /*Enables the voltage auto range
for period measurements.*/
SENSe:PERiod:VOLTage:RANGe:AUTO? /*Sets or queries the on/off
status of the voltage auto range for period measurements. The query
returns 1.*/
```

**3.17.43 [SENSe]:PERiod:SECondary****Syntax**

```
[SENSe]:PERiod:SECondary <second>
```

```
[SENSe]:PERiod:SECondary?
```

**Description**

Sets or queries the secondary measurement function for period measurement.

**Parameter**

Name	Type	Range	Default
<second>	Discrete	{"OFF" "CALCulate:DATA" "VOLTage:AC"}	"OFF"

**Remarks**

- **"OFF"**: disables the secondary measurement function.
- **"CALCulate:DATA"**: the measurement value prior to any maths operations (including relative operation).
- **"VOLTage:AC"**: AC voltage measurement.

**Return Format**

The query returns "OFF", "CALC:DATA", or "VOLT:AC".

**Examples**

```
SENSe:PERiod:SECondary "VOLTage:AC" /*Sets the secondary
measurement function to AC voltage measurement for period
measurement.*/
SENSe:PERiod:SECondary? /*Queries the secondary measurement
function for period measurement. The query returns "VOLT:AC".*/
```

### 3.17.44 [SENSe]:RESistance:NPLC

**Syntax**

[SENSe]:RESistance:NPLC {<plc>|<lim>}

[SENSe]:RESistance:NPLC?

**Description**

Sets or queries the integration time in power-line cycles (PLCs) for 2-wire resistance measurements.

**Parameter**

Name	Type	Range	Default
<plc>	Real	{0.4 5 20}	20
<lim>	Discrete	{MIN MAX DEF}	-

**Remarks**

- Integration time is the period that the instrument's analog-to-digital (A/D) converter samples the input signal for a measurement. A longer integration time provides higher measurement resolution but slower measurement.

- The unit for integration time is PLC. 1PLC = 0.02 s.
- Setting the integration time also affects the measurement resolution. To see the relationship between integration time and resolution, see [Table 3.14 Resolution, Measurement Speed, and Integration Time](#).

### Return Format

The query returns the multiple of the current PLC in scientific notation.

### Examples

```
SENSe:RESistance:NPLC 5 /*Sets the integration time to 5PLC for 2-
wire resistance measurements.*/
SENSe:RESistance:NPLC? /*Queries the integration time for 2-wire
resistance measurements. The query returns 5.00000000E+00.*/
```

## 3.17.45 [SENSe]:RESistance:NULL[:STATe]

### Syntax

[SENSe]:RESistance:NULL[:STATe] <bool>

[SENSe]:RESistance:NULL[:STATe]?

### Description

Sets or queries the on/off status of the relative function for 2-wire resistance measurements.

### Parameter

Name	Type	Range	Default
<bool>	Bool	{0 1 OFF ON}	0 OFF

### Remarks

When the relative operation is enabled, the displayed reading value is the result obtained by subtracting the set relative value from the actual measurement value. That is, reading value = actual measurement value - relative value. You can use [\[SENSe\]:RESistance:NULL:VALue](#) to set the relative value.

### Return Format

The query returns 0 or 1.

### Examples

```
SENSe:RESistance:NULL:STATe ON /*Enables the relative function for
2-wire resistance measurements.*/
SENSe:RESistance:NULL:STATe? /*Queries whether the relative
function is enabled for 2-wire resistance measurements. The query
returns 1.*/
```

### 3.17.46 [SENSe]:RESistance:NULL:VALue

#### Syntax

```
[SENSe]:RESistance:NULL:VALue {<value>|<lim>}
```

```
[SENSe]:RESistance:NULL:VALue?
```

#### Description

Sets or queries the relative value for 2-wire resistance measurements.

#### Parameter

Name	Type	Range	Default
<value>	Real	Refer to <i>Remarks</i>	-
<lim>	Discrete	{MIN MAX DEF}	-

#### Remarks

The range of the relative value is -120% to +120% of the current 2-wire resistance measurement range (*[SENSe]:RESistance:RANGe*).

#### Return Format

The query returns the relative value in scientific notation.

#### Examples

```
SENSe:RESistance:NULL:VALue 100 /*Sets the relative value to 100 Ω
for 2-wire resistance measurements.*/
SENSe:RESistance:NULL:VALue? /*Queries the relative value for 2-
wire resistances measurements. The query returns 1.00000000E+02.*/
```

### 3.17.47 [SENSe]:RESistance:NULL:VALue:AUTO

#### Syntax

```
[SENSe]:RESistance:NULL:VALue:AUTO <bool>
```

```
[SENSe]:RESistance:NULL:VALue:AUTO?
```

#### Description

Sets or queries the on/off status of the auto relative function for 2-wire resistance measurements.

#### Parameter

Name	Type	Range	Default
<bool>	Bool	{0 1 OFF ON}	1 ON

**Remarks**

- When the auto relative function is enabled, the relative value is automatically set to the the first 2-wire resistance measurement value.
- When `[SENSe]:RESistance:NULL:VALue` is used to specify a valid relative value, the auto relative function will be disabled.

**Return Format**

The query returns 0 or 1.

**Examples**

```
SENSe:RESistance:NULL:VALue:AUTO ON /*Enables the auto relative
function for 2-wire resistance measurements.*/
SENSe:RESistance:NULL:VALue:AUTO? /*Queries whether the auto
relative function is enabled for 2-wire resistance measurements.
The query returns 1.*/
```

### 3.17.48 [SENSe]:RESistance:RANGe

**Syntax**

`[SENSe]:RESistance:RANGe {<range>|<lim>}`

`[SENSe]:RESistance:RANGe?`

**Description**

Sets or queries the range for 2-wire resistance measurements.

**Parameter**

Name	Type	Range	Default
<range>	Discrete	{100Ω 1kΩ 10kΩ 100kΩ 1MΩ 10MΩ 50MΩ AUTO}	AUTO
<lim>	Discrete	{MIN MAX DEF}	-

**Remarks**

- The default unit of <range> is Ω. You can set the value in scientific notation, for example, 1E+3 (1 kΩ).
- AUTO (auto range) automatically selects the range according to the input signal. You can use `[SENSe]:RESistance:RANGe:AUTO` to set the 2-wire resistance measurement to auto range.

**Return Format**

The query returns the range for 2-wire resistance measurements in scientific notation.

### Examples

```
SENSe:RESistance:RANGe 100 /*Sets the range to 100 Ω for 2-wire
resistance measurements.*/
SENSe:RESistance:RANGe? /*Queries the range for 2-wire resistance
measurements. The query returns 1.00000000E+02.*/
```

## 3.17.49 [SENSe]:RESistance:RANGe:AUTO

### Syntax

```
[SENSe]:RESistance:RANGe:AUTO <bool>
```

```
[SENSe]:RESistance:RANGe:AUTO
```

### Description

Sets or queries the on/off status of auto range for 2-wire resistance measurements.

### Parameter

Name	Type	Range	Default
<bool>	Bool	{0 1 OFF ON}	1 ON

### Remarks

Auto range function automatically selects the range according to the input signal. After you use `[SENSe]:RESistance:RANGe` to select a fixed range, the auto range is automatically disabled.

### Return Format

The query returns 0 or 1.

### Examples

```
SENSe:RESistance:RANGe:AUTO ON /*Enables the auto range function
for 2-wire resistance measurements.*/
SENSe:RESistance:RANGe:AUTO? /*Queries whether the auto range
function is enabled for 2-wire resistance measurements. The query
returns 1.*/
```

## 3.17.50 [SENSe]:RESistance:RESolution

### Syntax

```
[SENSe]:RESistance:RESolution {<resolution>|<lim>}
```

```
[SENSe]:RESistance:RESolution?
```

### Description

Sets or queries the resolution for 2-wire resistance measurements.

**Parameter**

Name	Type	Range	Default
<resolution>	Real	Refer to <i>Table 3.14 Resolution, Measurement Speed, and Integration Time</i>	10 ppm x range
<lim>	Discrete	{MAX MIN DEF}	-

**Remarks**

The default unit of <resolution> is  $\Omega$ . You can set the parameters in scientific notation, for example, 1E+2 (100  $\Omega$ ).

**Return Format**

The query returns the resolution for 2-wire resistance measurements in scientific notation.

**Examples**

```
SENSe:RESistance:RESolution 100 /*Sets the resolution to 100  $\Omega$  for
2-wire resistance measurements.*/
SENSe:RESistance:RESolution? /*Queries the resolution for 2-wire
resistance measurements. The query returns 1.00000000E+02.*/
```

### 3.17.51 [SENSe]:RESistance:SECondary

**Syntax**

[SENSe]:RESistance:SECondary <second>

[SENSe]:RESistance:SECondary?

**Description**

Sets or queries the secondary measurement function for 2-wire resistance measurement.

**Parameter**

Name	Type	Range	Default
<second>	Discrete	{"OFF" "CALCulate:DATA"}	"OFF"

**Remarks**

- **"OFF"**: disables the secondary measurement function.
- **"CALCulate:DATA"**: the measurement value prior to any maths operations (including relative operation).



**Return Format**

The query returns "OFF" or "CALC:DATA".

**Examples**

```
SENSe:RESistance:SECondary "CALCulate:DATA" /*Sets the secondary
measurement function to the 2-wire resistance measurement without
any Maths operation for 2-wire resistance measurement.*/
SENSe:RESistance:SECondary? /*Queries the secondary measurement
function for 2-wire resistance measurement. The query returns
"CALC:DATA".*/
```

**3.17.52 [SENSe]:VOLTage:AC:NULL[:STATe]****Syntax**

[SENSe]:VOLTage:AC:NULL[:STATe] <bool>

[SENSe]:VOLTage:AC:NULL[:STATe]?

**Description**

Sets or queries the on/off status of the relative function for AC voltage measurements.

**Parameter**

Name	Type	Range	Default
<bool>	Bool	{0 1 OFF ON}	0 OFF

**Remarks**

When the relative operation is enabled, the displayed reading value is the result obtained by subtracting the set relative value from the actual measurement value. That is, reading value = actual measurement value - relative value. You can use [\[SENSe\]:VOLTage:AC:NULL:VALue](#) to set the relative value.

**Return Format**

The query returns 0 or 1.

**Examples**

```
SENSe:VOLTage:AC:NULL:STATe ON /*Enables the relative function for
AC voltage measurements.*/
SENSe:VOLTage:AC:NULL:STATe? /*Queries whether the relative
function is enabled for AC voltage measurements. The query returns
1.*/
```

**3.17.53 [SENSe]:VOLTage:AC:NULL:VALue****Syntax**

[SENSe]:VOLTage:AC:NULL:VALue {<value>|<lim>}

[SENSe]:VOLTage:AC:NULL:VALue?

### Description

Sets or queries the relative value for AC voltage measurements.

### Parameter

Name	Type	Range	Default
<value>	Real	Refer to <i>Remarks</i>	-
<lim>	Discrete	{MIN MAX DEF}	-

### Remarks

The range of the relative value is -120% to +120% of the current AC voltage measurement range (*[SENSe]:VOLTage:AC:RANGe*).

### Return Format

The query returns the relative value in scientific notation.

### Examples

```
SENSe:VOLTage:AC:NULL:VALue 10 /*Sets the relative value to 10 V
for AC voltage measurements.*/
SENSe:VOLTage:AC:NULL:VALue? /*Queries the relative value for AC
voltage measurements. The query returns 1.00000000E+01.*/
```

## 3.17.54 [SENSe]:VOLTage:AC:NULL:VALue:AUTO

### Syntax

[SENSe]:VOLTage:AC:NULL:VALue:AUTO <bool>

[SENSe]:VOLTage:AC:NULL:VALue:AUTO?

### Description

Sets or queries the on/off status of the auto relative function for AC voltage measurements.

### Parameter

Name	Type	Range	Default
<bool>	Bool	{0 1 OFF ON}	1 ON

### Remarks

- When the auto relative function is enabled, the relative value is automatically set to the the first AC voltage measurement value.
- When *[SENSe]:VOLTage:AC:NULL:VALue* is used to specify a valid relative value, the auto relative function will be disabled.

**Return Format**

The query returns 0 or 1.

**Examples**

```
SENSe:VOLTage:AC:NULL:VALue:AUTO ON /*Enables the auto relative
function for AC voltage measurements.*/
SENSe:VOLTage:AC:NULL:VALue:AUTO? /*Queries whether the auto
relative function is enabled for AC voltage measurements. The query
returns 1.*/
```

**3.17.55 [SENSe]:VOLTage:AC:RANGe****Syntax**

```
[SENSe]:VOLTage:AC:RANGe {<range>|<lim>}
```

```
[SENSe]:VOLTage:AC:RANGe?
```

**Description**

Sets or queries the range for AC voltage measurements.

**Parameter**

Name	Type	Range	Default
<range>	Discrete	{100mV 1V 10V 100V 750V  AUTO}	AUTO
<lim>	Discrete	{MIN MAX DEF}	-

**Remarks**

- The default unit of <range> is V. You can set the value in scientific notation, for example, 1E-3 (0.001 V). If the parameter is omitted, it is interpreted as the default value.
- AUTO (auto range) automatically selects the range according to the input signal. You can also use `[SENSe]:VOLTage:AC:RANGe:AUTO` to set the auto range for AC voltage measurements.

**Return Format**

The query returns the range for AC voltage measurements in scientific notation.

**Examples**

```
SENSe:VOLTage:AC:RANGe 1 /*Sets the range to 1 V for AC voltage
measurements.*/
SENSe:VOLTage:AC:RANGe? /*Queries the range for AC voltage
measurements. The query returns 1.00000000E+00.*/
```

### 3.17.56 [SENSe]:VOLTage:AC:RANGe:AUTO

#### Syntax

[SENSe]:VOLTage:AC:RANGe:AUTO <bool>

[SENSe]:VOLTage:AC:RANGe:AUTO?

#### Description

Sets or queries the on/off status of auto range for AC voltage measurements.

#### Parameter

Name	Type	Range	Default
<bool>	Bool	{0 1 OFF ON}	1 ON

#### Remarks

Auto range function automatically selects the range according to the input signal. After you use `[SENSe]:VOLTage:AC:RANGe` to select a fixed range, the auto range is automatically disabled.

#### Return Format

The query returns 0 or 1.

#### Examples

```
SENSe:VOLTage:AC:RANGe:AUTO ON /*Enables the auto range function
for AC voltage measurements.*/
SENSe:VOLTage:AC:RANGe:AUTO? /*Queries whether the auto range
function is enabled for AC voltage measurements. The query returns
1.*/
```

### 3.17.57 [SENSe]:VOLTage:AC:SECondary

#### Syntax

[SENSe]:VOLTage:AC:SECondary <second>

[SENSe]:VOLTage:AC:SECondary?

#### Description

Sets or queries the secondary measurement function for AC voltage measurement.

#### Parameter

Name	Type	Range	Default
<second>	Discrete	{"OFF" "CALCulate:DATA" "FREQuency" "PERiod"}	"OFF"

**Remarks**

- **"OFF"**: disables the secondary measurement function.
- **"CALCulate:DATA"**: the measurement value prior to any maths operations (including relative operation).
- **"FREQuency"**: frequency measurement.
- **"PERiod"**: period measurement.

**Return Format**

The query returns "OFF", "CALC:DATA", "FREQ", or "PER".

**Examples**

```
SENSe:VOLTage:AC:SECondary "FREQ" /*Sets the secondary measurement
function to frequency measurement for AC voltage measurement.*/
SENSe:VOLTage:AC:SECondary? /*Queries the secondary measurement
function for AC voltage measurement. The query returns "FREQ".*/
```

**3.17.58 [SENSe]:VOLTage:DC:NULL[:STATe]****Syntax**

[SENSe]:VOLTage:DC:NULL[:STATe] <bool>

[SENSe]:VOLTage:DC:NULL[:STATe]?

**Description**

Sets or queries the on/off status of the relative function for DC voltage measurements.

**Parameter**

Name	Type	Range	Default
<bool>	Bool	{0 1 OFF ON}	0 OFF

**Remarks**

When the relative function is enabled, the displayed reading value is the result obtained by subtracting the set relative value from the actual measurement value. That is, reading value = actual measurement value - relative value. You can use [\[SENSe\]:VOLTage:DC:NULL:VALue](#) to set the relative value.

**Return Format**

The query returns 0 or 1.

**Examples**

```
SENSe:VOLTage:DC:NULL:STATe ON /*Enables the relative function for
DC voltage measurements.*/
```

```
SENSe:VOLTage:DC:NULL:STATe? /*Queries whether the relative
function is enabled for DC voltage measurements. The query returns
1.*/
```

### 3.17.59 [SENSe]:VOLTage:DC:NULL:VALue

#### Syntax

```
[SENSe]:VOLTage:DC:NULL:VALue {<value>|<lim>}
```

```
[SENSe]:VOLTage:DC:NULL:VALue?
```

#### Description

Sets or queries the relative value for DC voltage measurements.

#### Parameter

Name	Type	Range	Default
<value>	Real	Refer to <i>Remarks</i>	-
<lim>	Discrete	{MIN MAX DEF}	-

#### Remarks

The range of the relative value is -120% to +120% of the current DC voltage measurement range (*[SENSe]:VOLTage:DC:RANGe*).

#### Return Format

The query returns the relative value in scientific notation.

#### Examples

```
SENSe:VOLTage:DC:NULL:VALue 0.001 /*Sets the relative value to 1 mV
for DC voltage measurements.*/
SENSe:VOLTage:DC:NULL:VALue? /*Queries the relative value for DC
voltage measurements. The query returns 1.00000000E-03.*/
```

### 3.17.60 [SENSe]:VOLTage:DC:NULL:VALue:AUTO

#### Syntax

```
[SENSe]:VOLTage:DC:NULL:VALue:AUTO <bool>
```

```
[SENSe]:VOLTage:DC:NULL:VALue:AUTO?
```

#### Description

Sets or queries the on/off status of the auto relative function for DC voltage measurements.

**Parameter**

Name	Type	Range	Default
<bool>	Bool	{0 1 OFF ON}	1 ON

**Remarks**

- When the auto relative function is enabled, the relative value is automatically set to the the first DC voltage measurement value.
- When `[SENSe]:VOLTage:DC:NULL:VALue` is used to specify a valid relative value, the auto relative function will be disabled.

**Return Format**

The query returns 0 or 1.

**Examples**

```
SENSe:VOLTage:DC:NULL:VALue:AUTO ON /*Enables the auto relative
function for DC voltage measurements.*/
SENSe:VOLTage:DC:NULL:VALue:AUTO? /*Queries whether the auto
relative function is enabled for DC voltage measurements. The query
returns 1.*/
```

### 3.17.61 [SENSe]:VOLTage:DC:RANGE

**Syntax**

`[SENSe]:VOLTage:DC:RANGE {<range>|<lim>}`

`[SENSe]:VOLTage:DC:RANGE?`

**Description**

Sets or queries the range for DC voltage measurements.

**Parameter**

Name	Type	Range	Default
<range>	Discrete	{100mV 1V 10V 100V 1000V AUTO}	AUTO
<lim>	Discrete	{MIN MAX DEF}	-

**Remarks**

- The default unit of <range> is V. You can set the value in scientific notation, for example, 1E-3 (0.001 V).
- AUTO (auto range) automatically selects the range according to the input signal. You can also use `[SENSe]:VOLTage:AC:RANGE:AUTO` to set the auto range for AC voltage measurements.

**Return Format**

The query returns the range for DC voltage measurements in scientific notation.

**Examples**

```
SENSe:VOLTage:DC:RANGe 1 /*Sets the range to 1 V for DC voltage
measurements.*/ SENSE:VOLTage:DC:RANGe? /*Queries the range for DC
voltage measurements. The query returns 1.00000000E+00.*/
```

**3.17.62 [SENSe]:VOLTage:DC:RANGe:AUTO****Syntax**

```
[SENSe]:VOLTage:DC:RANGe:AUTO <bool>
```

```
[SENSe]:VOLTage:DC:RANGe:AUTO?
```

**Description**

Sets or queries the on/off status of auto range for DC voltage measurements.

**Parameter**

Name	Type	Range	Default
<bool>	Bool	{0 1 OFF ON}	1 ON

**Remarks**

Auto range function automatically selects the range according to the input signal. After you use `[SENSe]:VOLTage:DC:RANGe` to select a fixed range, the auto range is automatically disabled.

**Return Format**

The query returns 0 or 1.

**Examples**

```
SENSe:VOLTage:DC:RANGe:AUTO ON /*Enables the auto range function
for DC voltage measurements.*/
SENSe:VOLTage:DC:RANGe:AUTO? /*Queries whether the auto range
function is enabled for DC voltage measurements. The query returns
1.*/
```

**3.17.63 [SENSe]:VOLTage[:DC]:NPLC****Syntax**

```
[SENSe]:VOLTage[:DC]:NPLC {<plc>|<lim>}
```

```
[SENSe]:VOLTage[:DC]:NPLC?
```



**Description**

Sets or queries the integration time in power-line cycles (PLCs) for DC voltage measurements.

**Parameter**

Name	Type	Range	Default
<plc>	Real	{0.4 5 20}	20
<lim>	Discrete	{MIN MAX DEF}	-

**Remarks**

- Integration time is the period that the instrument's analog-to-digital (A/D) converter samples the input signal for a measurement. A longer integration time provides higher measurement resolution but slower measurement.
- The unit for integration time is PLC. 1PLC = 0.02 s.
- Setting the integration time also affects the measurement resolution. To see the relationship between integration time and resolution, see [Table 3.14 Resolution, Measurement Speed, and Integration Time](#).

**Return Format**

The query returns the multiple of the current PLC in scientific notation.

**Examples**

```
SENSe:VOLTage:DC:NPLC 5 /*Sets the integration time to 5PLC for DC
voltage measurements.*/
SENSe:VOLTage:DC:NPLC? /*Queries the integration time for DC
voltage measurements. The query returns 5.00000000E+00.*/
```

**3.17.64 [SENSe]:VOLTage[:DC]:RESolution****Syntax**

```
[SENSe]:VOLTage[:DC]:RESolution {<resolution>|<lim>}
```

```
[SENSe]:VOLTage[:DC]:RESolution?
```

**Description**

Sets or queries the resolution for DC voltage measurements.

**Parameter**

Name	Type	Range	Default
<resolution>	Real	Refer to <a href="#">Table 3.14 Resolution, Measurement Speed, and Integration Time</a>	10 ppm x range

Name	Type	Range	Default
<lim>	Discrete	{MAX MIN DEF}	-

**Remarks**

The default unit of <resolution> is V. You can set the parameters in scientific notation, for example, 1E-4 (0.1 mV).

**Return Format**

The query returns the resolution for DC voltage measurements in scientific notation.

**Examples**

```
SENSe:VOLTage:DC:RESolution 1E-4 /*Sets the resolution to 0.1 mV
for DC voltage measurements.*/
SENSe:VOLTage:DC:RESolution? /*Queries the resolution for DC
voltage measurements. The query returns 1.00000000E-04.*/
```

### 3.17.65 [SENSe]:VOLTage[:DC]:SECondary

**Syntax**

[SENSe]:VOLTage[:DC]:SECondary <second>

[SENSe]:VOLTage[:DC]:SECondary?

**Description**

Sets or queries the secondary measurement function for DC voltage measurement.

**Parameter**

Name	Type	Range	Default
<second>	Discrete	{"OFF" "CALCulate:DATA"}	"OFF"

**Remarks**

- **"OFF"**: disables the secondary measurement function.
- **"CALCulate:DATA"**: the measurement value prior to any maths operations (including relative operation).

**Return Format**

The query returns "OFF" or "CALC:DATA".

**Examples**

```
SENSe:VOLTage:DC:SECondary "CALCulate:DATA" /*Sets the secondary
measurement function to DC voltage measurement without any Maths
operation for DC voltage measurement.*/
SENSe:VOLTage:DC:SECondary? /*Queries the secondary measurement
```

```
function for DC voltage measurement. The query returns
"CALC:DATA".*/
```

### 3.18 STATus Commands

STATus commands are used to set and query the Questionable Data register and Operation Status register.

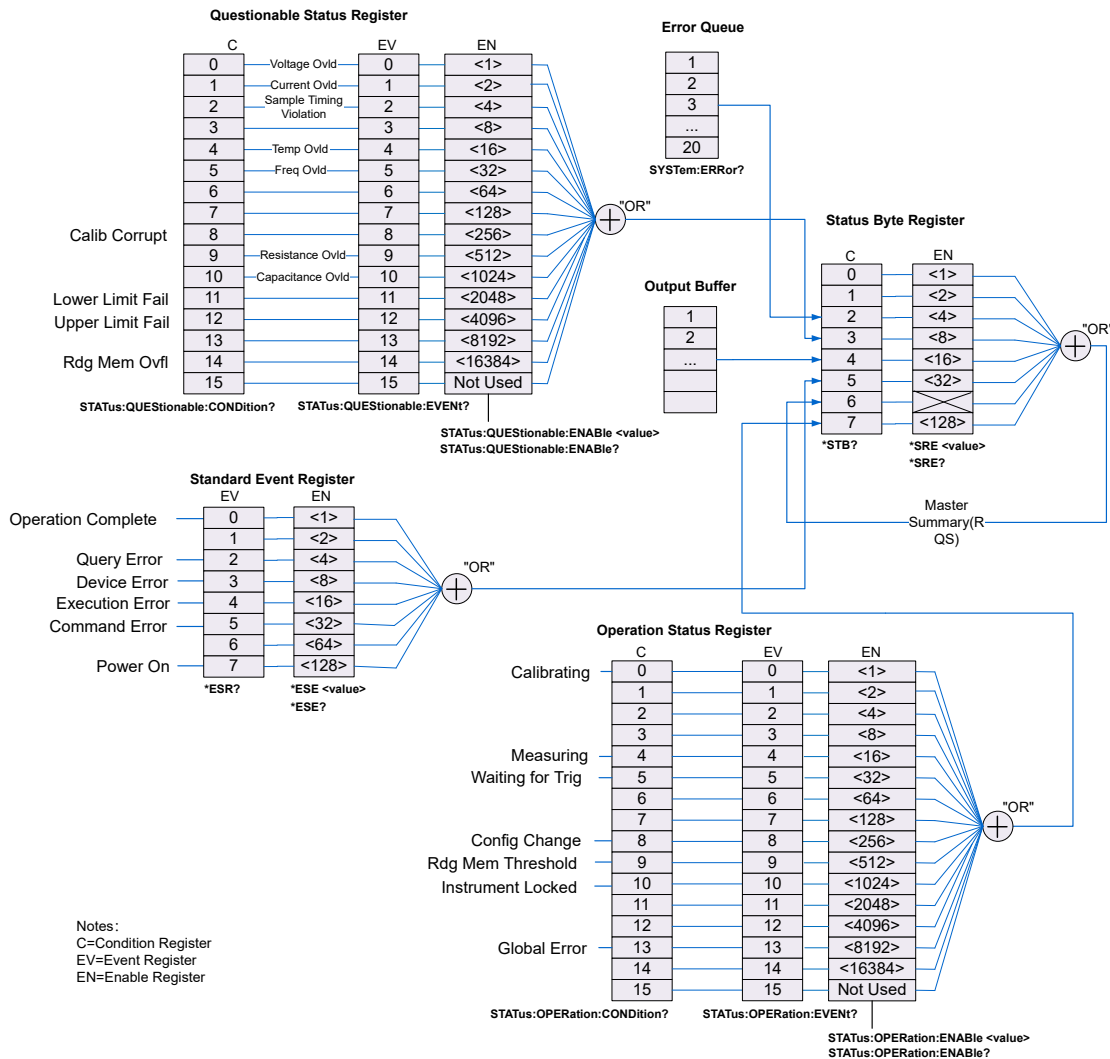


Figure 3.1 DM858/DM858E Status System

The following table lists the bit definitions for the Questionable Data register:

**Table 3.118 Bit Definitions for the Questionable Data Register**

Bit	Bit Name	Decimal Value	Definition
0	Voltage Overload	1	Only reported as event. Returns 0. Read the Event Register.
1	Current Overload	2	Only reported as event. Returns 0. Read the Event Register.
2	Sample Timing Violation	4	A sample timing violation has occurred. Sample timing may not be valid.
3	Not Used	8	Reserved for future use
4	Temperature Overload	16	Only reported as event. Returns 0. Read the Event Register.
5	Frequency Overload/ Underflow	32	Only reported as event. Returns 0. Read the Event Register.
6	Not Used	64	Reserved for future use
7	Not Used	128	Reserved for future use
8	Calibration Error	256	Error occurs in calibration.
9	Resistance Overload	512	Only reported as event. Returns 0. Read the Event Register.
10	Capacitance Overload	1024	Only reported as event. Returns 0. Read the Event Register.
11	Lower Limit Failed	2048	The most recent measurement failed the lower limit test.
12	Upper Limit Failed	4096	The most recent measurement failed the upper limit test.
13	Not Used	8192	Reserved for future use
14	Memory Overflow	16384	Reading memory is full. One or more (oldest) measurements have been lost.

Bit	Bit Name	Decimal Value	Definition
15	Not Used	32768	Reserved for future use

The following table lists the bit definitions for the Operation Status register:

**Table 3.119 Bit Definitions for the Operation Status Register**

Bit	Bit Name	Decimal Value	Definition
0	Calibrating	1	The instrument is performing a calibration.
1	Not Used	2	Reserved for future use
2	Not Used	4	Reserved for future use
3	Not Used	8	Reserved for future use
4	Measuring	16	The instrument is initiated, and is making or about to make a measurement.
5	Waiting for Trigger	32	The instrument is waiting for a trigger.
6	Not Used	64	Reserved for future use
7	Not Used	128	Reserved for future use
8	Configuration change	256	The instrument configuration has been changed since the last measurement and reading, either from the front panel or from SCPI.
9	Memory Threshold	512	Preset number of measurements have been stored in measurement memory.
10	Instrument Locked	1024	Set if the instrument is locked. Cleared when the lock is released.
11	Not Used	2048	Reserved for future use
12	Not Used	4096	Reserved for future use
13	Global Error	8192	Set if any remote interface has an error in its error queue; cleared otherwise.
14	Not Used	16384	Reserved for future use

Bit	Bit Name	Decimal Value	Definition
15	Not Used	32768	Returns 0.

### 3.18.1 STATus:OPERation:CONDition?

#### Syntax

```
STATus:OPERation:CONDition?
```

#### Description

Queries the condition register of the Operation Status register.

#### Parameter

None.

#### Remarks

The query returns the sum of the bits in the condition register of the Operation Status register. The register is read-only and the bits are not cleared when you read the register.

#### Return Format

The query returns a decimal value, which corresponds to the binary-weighted sum of the bits to be enabled in the enable register. For example, the query might return +32 (bit 5).

#### Examples

```
STATus:OPERation:CONDition? /*Queries the condition register of the
Operation Status register. The query returns +32.*/
```

### 3.18.2 STATus:OPERation:ENABLE

#### Syntax

```
STATus:OPERation:ENABLE <value>
```

```
STATus:OPERation:ENABLE?
```

#### Description

Queries the enable register of the Operation Status register.

#### Parameter

Name	Type	Range	Default
<value>	Integer	Refer to <i>Remarks</i>	-

**Remarks**

- <value> is a decimal value, which corresponds to the binary-weighted sum of the bits to be enabled in the register. For example, to enable bit 5 (decimal value = 32) and bit 9 (decimal value = 512), the corresponding decimal value would be 544 (32 + 512).
- The enable register can specify which bits in the event register will be reported to the "Status Byte" register group. The enable register is a read/write register.
- *STATUS:PRESet* will clear all bits in the enable register.

**Return Format**

The query returns a decimal value, which corresponds to the binary-weighted sum of the bits in the register. For example, the query might return +32.

**Examples**

```
STATUS:OPERation:ENABle 32 /*Enables bit 5 (32) in the enable
register of the Questionable Data register.*/
STATUS:OPERation:ENABle? /*Queries the bit to be enabled in the
enable register. The query returns +32.*/
```

### 3.18.3 STATUS:OPERation[:EVENT]?

**Syntax**

```
STATUS:OPERation[:EVENT]?
```

**Description**

Queries the event register of the Operation Status register.

**Parameter**

None.

**Remarks**

Executes this command and the query returns a decimal value (corresponding to the binary-weighted sum of all bits set in the register) and clear the status of the register.

**Return Format**

The query returns a decimal value, which corresponds to the binary-weighted sum of the bits in the event register of the Operation Status register. For example, the query might return +32.

**Examples**

```
STATUS:OPERation:EVENT? /*Queries the event register of the
Operation Status register. The query returns +32.*/
```

### 3.18.4 STATus:PRESet

---

**Syntax**

STATus:PRESet

**Description**

Clears all bits in the Operation Status enable register and the Questionable Data enable register.

**Parameter**

None.

**Remarks**

None.

**Return Format**

None.

**Examples**

```
STATus:PRESet /*Clears all bits in the Operation Status enable register and the Questionable Data enable register.*/
```

### 3.18.5 STATus:QUESTIONable:CONDition?

---

**Syntax**

STATus:QUESTIONable:CONDition?

**Description**

Queries the condition register of the Questionable Data register.

**Parameter**

None.

**Remarks**

The query returns the sum of the bits in the condition register of the Questionable Data register. The register is read-only and the bits are not cleared when you read the register.

**Return Format**

The query returns a decimal value, which corresponds to the binary-weighted sum of the bits to be enabled in the enable register. For example, the query might return +4096 (bit 12).



### Examples

```
STATUS:QUESTIONable:CONDition? /*Queries the condition register of
the Questionable Data register. The query returns +4096.*/
```

## 3.18.6 STATus:QUESTIONable:ENABLE

### Syntax

```
STATUS:QUESTIONable:ENABLE <value>
```

```
STATUS:QUESTIONable:ENABLE?
```

### Description

Sets or queries the enable register of the Questionable Data register.

### Parameter

Name	Type	Range	Default
<value>	Integer	Refer to <i>Remarks</i>	-

### Remarks

- <value> is a decimal value, which corresponds to the binary-weighted sum of the bits to be enabled in the register. For example, to enable bit 0 (decimal value = 1), bit 1 (decimal value = 2), and bit 12 (decimal value = 4096), the corresponding decimal value would be 4099 (1 + 2 + 4096).
- The enable register can specify which bits in the event register will be reported to the "Status Byte" register group. The enable register is a read/write register.
- *STATUS:PRESet* will clear all bits in the enable register.

### Return Format

The query returns a decimal value, which corresponds to the binary-weighted sum of the bits to be enabled in the enable register. For example, the query might return +512 (bit 9).

### Examples

```
STATUS:QUESTIONable:ENABLE 512 /*Enables bit 9 (512) in the enable
register of the Questionable Data register.*/
STATUS:QUESTIONable:ENABLE? /*Queries the bit to be enabled in the
enable register. The query returns +512.*/
```

## 3.18.7 STATus:QUESTIONable[:EVENT]?

### Syntax

```
STATUS:QUESTIONable[:EVENT]?
```

**Description**

Queries the event register of the Questionable Data register.

**Parameter**

None.

**Remarks**

Executes this command and the query returns a decimal value (corresponding to the binary-weighted sum of all bits set in the register) and clear the status of the register.

**Return Format**

The query returns a decimal value, which corresponds to the binary-weighted sum of the bits set in the event register. For example, the query might return +1024.

**Examples**

```
STATus:QUEStionable:EVENT? /*Queries the event register of the  
Questionable Data register. The query returns +1024.*/
```

## 3.19 SYSTem Commands

**SYSTem** commands are used to set or query the beeper on/off state, network status, and system information.

### 3.19.1 SYSTem:BEEPer[:IMMediate]

**Syntax**

```
SYSTem:BEEPer[:IMMediate]
```

**Description**

Issues a single beep immediately.

**Parameter**

None.

**Remarks**

None.

**Return Format**

None.

**Examples**

```
SYSTem:BEEPer:IMMediate /*Issues a single beep immediately.*/
```

### 3.19.2 SYSTem:BEEPer:STATe

#### Syntax

```
SYSTem:BEEPer:STATe <bool>
```

```
SYSTem:BEEPer:STATe?
```

#### Description

Sets or queries the on/off status of the beeper.

#### Parameter

Name	Type	Range	Default
<bool>	Bool	{0 1 OFF ON}	-

#### Remarks

When the beeper is enabled, the instrument enables the click sound when the touch screen is used or when an error is generated from the remote control.

#### Return Format

The query returns 0 or 1.

#### Examples

```
SYSTem:BEEPer:STATe ON /*Enables the beeper.*/
SYSTem:BEEPer:STATe? /*Queries whether the beeper is enabled. The
query returns 1.*/
```

### 3.19.3 SYSTem:COMMunicate:LAN:AUTOip

#### Syntax

```
SYSTem:COMMunicate:LAN:AUTOip <bool>
```

```
SYSTem:COMMunicate:LAN:AUTOip?
```

#### Description

Sets or queries the on/off status of the Auto IP mode.

#### Parameter

Name	Type	Range	Default
<bool>	Bool	{0 1 OFF ON}	1 ON

#### Remarks

- Before using the LAN interface, use the network cable to connect the instrument to the PC or the local area network (LAN) where the PC resides.

- This instrument provides three IP configuration types: DHCP, Auto IP, and Static IP.
- In "Auto IP" mode, the instrument will acquire the IP address ranging from "169.254.0.1" to "169.254.255.254" and the subnet mask (255.255.0.0) automatically based on the current network configuration.
- When the three IP configuration types are all turned on, the priority of the parameter configuration from high to low is "DHCP", "Auto IP", and "Static IP". Therefore, to enable the Auto IP mode, DHCP (*SYSTem:COMMunicate:LAN:DHCP*) should be disabled.
- The three IP configuration types cannot be all turned off at the same time.
- After sending this command, execute *SYSTem:COMMunicate:LAN:UPDate* to apply the current network setting for the new parameters to take effect.

#### Return Format

The query returns 0 or 1.

#### Examples

```
SYSTem:COMMunicate:LAN:AUTOip ON /*Enables the Auto IP mode.*/  
SYSTem:COMMunicate:LAN:AUTOip? /*Queries the on/off status of the  
Auto IP mode.*/
```

### 3.19.4 SYSTem:COMMunicate:LAN:CONTRol?

#### Syntax

```
SYSTem:COMMunicate:LAN:CONTRol?
```

#### Description

Reads the port number of the initial control connecting port for socket communication.

#### Parameter

None.

#### Remarks

You can use the Socket Control connection to send a Device Clear (DCL) to the instrument or to detect pending Service Request (SRQ) events.

#### Return Format

The query returns the port number; if the interface does not support the socket, it returns 0.

### Examples

```
SYSTem:COMMunicate:LAN:CONTRol? /*Reads the port number of the
initial control connecting port for socket communication. The query
returns 5025.*/
```

## 3.19.5 SYSTem:COMMunicate:LAN:DHCP

### Syntax

```
SYSTem:COMMunicate:LAN:DHCP <bool>
```

```
SYSTem:COMMunicate:LAN:DHCP?
```

### Description

Sets or queries the on/off status of the DHCP mode.

### Parameter

Name	Type	Range	Default
<bool>	Bool	{0 1 OFF ON}	1 ON

### Remarks

- When the three IP configuration types are all turned on, the priority of the parameter configuration from high to low is "DHCP", "Auto IP", and "Static IP". The three configuration types cannot be all turned off at the same time.
- In DHCP mode, it automatically assigns an IP address, subnet mask, and default gateway to the instrument from a DHCP server.
- After sending this command, execute *SYSTem:COMMunicate:LAN:UPDate* to apply the current network setting for the new parameters to take effect.

### Return Format

The query returns 0 or 1.

### Examples

```
SYSTem:COMMunicate:LAN:DHCP ON /*Enables the DHCP mode.*/
SYSTem:COMMunicate:LAN:DHCP? /*Queries whether the DHCP mode is
enabled. The query returns 1.*/
```

## 3.19.6 SYSTem:COMMunicate:LAN:DNS

### Syntax

```
SYSTem:COMMunicate:LAN:DNS "<dns>"
```

```
SYSTem:COMMunicate:LAN:DNS?
```

### Description

Sets or queries the DNS address for the LAN interface.

**Parameter**

Name	Type	Range	Default
<dns>	ASCII string	Refer to <i>Remarks</i>	-

**Remarks**

- The parameter of <dns> should be enclosed by double quotation marks. The format of "<dns>" is "nnn.nnn.nnn.nnn"; wherein, the first nnn ranges from 1 to 223 (excluding 127), and the other three range from 0 to 255.
- When Static IP mode is enabled with both DHCP and Auto IP disabled, this command is valid.
- After sending this command, execute *SYSTem:COMMunicate:LAN:UPDate* to apply the current network setting for the new parameters to take effect.

**Return Format**

The query returns a string enclosed by double quotation marks, for example, "172.16.3.2".

**Examples**

```
SYSTem:COMMunicate:LAN:DNS "172.16.3.2" /*Sets the DNS address to
172.16.3.2.*/
SYSTem:COMMunicate:LAN:DNS? /*Queries the DNS address of the
instrument. The query returns "172.16.3.2".*/
```

### 3.19.7 SYSTem:COMMunicate:LAN:GATeway

**Syntax**

```
SYSTem:COMMunicate:LAN:GATeway "<gateway>"
```

```
SYSTem:COMMunicate:LAN:GATeway?
```

**Description**

Sets or queries the default gateway for the LAN interface.

**Parameter**

Name	Type	Range	Default
<gateway>	ASCII string	Refer to <i>Remarks</i>	-

**Remarks**

- The parameter of <gateway> should be enclosed by double quotation marks. The format of "<gateway>" is "nnn.nnn.nnn.nnn"; wherein, the first nnn ranges from 1 to 223 (excluding 127), and the other three range from 0 to 255.

- When Static IP mode is enabled with both DHCP and Auto IP disabled, this command is valid.
- After sending this command, execute `SYSTem:COMMunicate:LAN:UPDate` to apply the current network setting for the new parameters to take effect.

#### Return Format

The query returns a string enclosed by double quotation marks, for example, "172.16.3.1".

#### Examples

```
SYSTem:COMMunicate:LAN:GATeway "172.16.3.1" /*Sets the default gateway to 172.16.3.1.*/
SYSTem:COMMunicate:LAN:GATeway? /*Queries the default gateway of the instrument. The query returns "172.16.3.1".*/
```

### 3.19.8 SYSTem:COMMunicate:LAN:HOSTname

#### Syntax

```
SYSTem:COMMunicate:LAN:HOSTname "<name>"
```

```
SYSTem:COMMunicate:LAN:HOSTname?
```

#### Description

Sets or queries the hostname of the instrument.

#### Parameter

Name	Type	Range	Default
<name>	ASCII string	Refer to <i>Remarks</i>	-

#### Remarks

- The parameter of <name> should be enclosed by double quotation marks. <name> is the specified hostname with a maximum length of 15 characters. The name must start with the letter from A to Z. It can contain letters, numbers (0-9), and hyphens (-).
- This command is valid only when mDNS is enabled.

#### Return Format

The query returns a string enclosed by double quotation marks.

#### Examples

```
SYSTem:COMMunicate:LAN:HOSTname "RIGOL123" /*Sets the hostname to "RIGOL123".*/
SYSTem:COMMunicate:LAN:HOSTname? /*Queries the hostname. The query returns "RIGOL123".*/
```

### 3.19.9 SYSTem:COMMunicate:LAN:IPADdress

#### Syntax

```
SYSTem:COMMunicate:LAN:IPADdress "<ip>"
```

```
SYSTem:COMMunicate:LAN:IPADdress?
```

#### Description

Sets or queries the IP address for the LAN interface.

#### Parameter

Name	Type	Range	Default
<ip>	ASCII string	Refer to <i>Remarks</i>	-

#### Remarks

- The parameter of <ip> should be enclosed by double quotation marks. The format of "<ip>" is "nnn.nnn.nnn.nnn"; wherein, the first nnn ranges from 1 to 223 (excluding 127), and the other three range from 0 to 255.
- When Static IP mode is enabled with both DHCP and Auto IP disabled, this command is valid.
- After sending this command, execute *SYSTem:COMMunicate:LAN:UPDate* to apply the current network setting for the new parameters to take effect.

#### Return Format

The query returns a string enclosed by double quotation marks, for example, "169.254.149.35".

#### Examples

```
SYSTem:COMMunicate:LAN:IPADdress "169.254.149.35" /*Sets the IP
address to 169.254.149.35.*/
SYSTem:COMMunicate:LAN:IPADdress? /*Queries the IP address of the
instrument. The query returns "169.254.149.35".*/
```

### 3.19.10 SYSTem:COMMunicate:LAN:MAC?

#### Syntax

```
SYSTem:COMMunicate:LAN:MAC?
```

#### Description

Queries the MAC address of the instrument.

#### Parameter

None.



**Remarks**

A MAC (Media Access Control) address, also referred to as a hardware address, is used to identify the network device address. For each instrument, the MAC address is unique. When assigning the IP address for the instrument, the system uses the MAC address to identify the instrument. The MAC address (48 bits, namely 6 bytes) is usually expressed in hexadecimal format, such as 00:2A:A0:AA:E0:56.

**Return Format**

The query returns a string, for example, 00:2A:A0:AA:E0:56.

**Examples**

```
SYSTem:COMMunicate:LAN:MAC? /*Queries the MAC address of the instrument. The query might return 00:2A:A0:AA:E0:56.*/
```

### 3.19.11 SYSTem:COMMunicate:LAN:MANuip

**Syntax**

```
SYSTem:COMMunicate:LAN:MANuip <bool>
```

```
SYSTem:COMMunicate:LAN:MANuip?
```

**Description**

Sets or queries the on/off status of the Static IP mode.

**Parameter**

Name	Type	Range	Default
<bool>	Bool	{0 1 OFF ON}	0 OFF

**Remarks**

- In Static IP mode, you can self-define the network parameters (e.g. IP address) of the instrument.
- When the three IP configuration types are all turned on, the priority of the parameter configuration from high to low is "DHCP", "Auto IP", and "Static IP". Therefore, to enable the Static IP mode, you should disable the DHCP (*SYSTem:COMMunicate:LAN:DHCP*) and Auto IP (*SYSTem:COMMunicate:LAN:AUTOip*).
- The three IP configuration types cannot be all turned off at the same time.
- After sending this command, execute *SYSTem:COMMunicate:LAN:UPDate* to apply the current network setting for the new parameters to take effect.

**Return Format**

The query returns 0 or 1.

### Examples

```
SYSTem:COMMunicate:LAN:MANuip ON /*Enables the Static IP mode.*/
SYSTem:COMMunicate:LAN:MANuip? /*Queries whether the Static IP mode
is enabled. The query returns 1.*/
```

## 3.19.12 SYSTem:COMMunicate:LAN:SMASK

### Syntax

```
SYSTem:COMMunicate:LAN:SMASK "<smask>"
```

```
SYSTem:COMMunicate:LAN:SMASK?
```

### Description

Sets or queries the subnet mask for the LAN interface.

### Parameter

Name	Type	Range	Default
<smask>	ASCII string	Refer to <i>Remarks</i>	-

### Remarks

- The parameter of <smask> should be enclosed by double quotation marks. The format of "<smask>" is "nnn.nnn.nnn.nnn"; wherein, the range of "nnn" is from 0 to 255. A value of "0.0.0.0" or "255.255.255.255" indicates that subnet is not being used.
- When Static IP mode is enabled with both DHCP and Auto IP disabled, this command is valid.
- After sending this command, execute *SYSTem:COMMunicate:LAN:UPDate* to apply the current network setting for the new parameters to take effect.

### Return Format

The query returns a string enclosed by double quotation marks, for example, "255.255.255.0".

### Examples

```
SYSTem:COMMunicate:LAN:SMASK "255.255.255.0" /*Sets the subnet mask
to 255.255.255.0.*/
SYSTem:COMMunicate:LAN:SMASK? /*Queries the subnet mask. The query
returns "255.255.255.0".*/
```

## 3.19.13 SYSTem:COMMunicate:LAN:UPDate

### Syntax

```
SYSTem:COMMunicate:LAN:UPDate
```

**Description**

Stores any modifications of the LAN setting to the non-volatile memory and restarts the LAN drive program using the updated setting.

**Parameter**

None.

**Remarks**

- You must send this command after modifying the DHCP, DNS, gateway, IP address, and subnet mask settings.
- Complete all the LAN settings before sending this command.

**Return Format**

None.

**Examples**

None.

**3.19.14 SYSTem:DATE****Syntax**

**SYSTem:DATE** <year>,<month>,<day>

**SYSTem:DATE?**

**Description**

Sets or queries the date of the real time clock.

**Parameter**

Name	Type	Range	Default
<year>	Integer	2000 to 2100	-
<month>	Integer	1 to 12	-
<day>	Integer	1 to 31	-

**Remarks**

- The real time clock is used for the file timestamp for MMEMory system.
- You can use *SYSTem:TIME* to set the time.

**Return Format**

The query returns a string, for example, +2023,+07,+26.

**Examples**

```
SYSTem:DATE 2023,7,26 /*Sets the system date to July 26, 2023.*/
SYSTem:DATE? /*Queries the system date. The query returns
+2023,+07,+26.*/
```

**3.19.15 SYSTem:ERRor?****Syntax**

**SYSTem:ERRor?**

**Description**

Queries the error message in the error queue.

**Parameter**

None.

**Remarks**

- Up to 20 errors can be stored in the instrument's error queue. If more than 20 errors have occurred, the last error in the queue will be replaced with "-350", "Error queue overflow". No additional errors are stored until you remove errors from the queue.
- Errors are retrieved and cleared in first-in-first-out (FIFO) order.
- The error queue is cleared when power has been off or after *\*CLS* has been executed. The *\*RST* command does not clear the error queue.

**Return Format**

The query returns the error code and error message. For example, the query might return -224,"Illegal parameter value". If no errors have occurred, the query returns +0,"No error".

**Examples**

None.

**3.19.16 SYSTem:TIME****Syntax**

**SYSTem:TIME** <hour>,<minute>,<second>

**SYSTem:TIME?**

**Description**

Sets or queries the real time clock of the instrument.

**Parameter**

Name	Type	Range	Default
<hour>	Integer	0 to 23	-
<minute>	Integer	0 to 59	-
<second>	Integer	0 to 60	-

**Remarks**

- The real time clock is used for the file timestamp for MMEMory system.
- You can use *SYSTem:DATE* to set the date.

**Return Format**

The query returns a string, for example, 20,15,30.

**Examples**

```
SYSTem:TIME 20,15,30 /*Sets the real time clock to 20:15:30.*/
SYSTem:TIME? /*Queries the real time clock. The query returns
20,15,30.*/
```

### 3.19.17 SYSTem:VERSion?

**Syntax**

**SYSTem:VERSion?**

**Description**

Queries the current SCPI version number.

**Parameter**

None.

**Remarks**

None.

**Return Format**

The query returns a string (SCPI version) in the form of "YYYY.V", where "YYYY" represents the year of the version and "V" represents a version for that year (e.g. 1999.0).

**Examples**

```
SYSTem:VERSion? /*Queries the current SCPI version number. The
query returns 1999.0.*/
```

## 3.20 TRIGger Commands

TRIGger commands are used to set or query the trigger count and trigger source.

### 3.20.1 TRIGger:COUNT

#### Syntax

```
TRIGger:COUNT {<count>|<lim>}
```

```
TRIGger:COUNT?
```

#### Description

Sets or queries the number of triggers accepted in Single (Bus) trigger.

#### Parameter

Name	Type	Range	Default
<count>	Integer	1 to 1000	1
<lim>	Discrete	{MIN MAX DEF}	-

#### Remarks

- You can use the specified trigger count in conjunction with the sample count per trigger (*SAMPLE:COUNT*). In this case, the number of readings for single trigger will be the product of the sample count per trigger and trigger count.
- The set trigger count is valid only in Single trigger (*TRIGger:SOURce*).
- You can only set the trigger count in remote mode.

#### Return Format

The query returns an integer.

#### Examples

```
TRIGger:COUNT 50 /*Sets the trigger count to 50.*/
TRIGger:COUNT? /*Queries the trigger count. The query returns 50.*/
```

### 3.20.2 TRIGger:SOURce

#### Syntax

```
TRIGger:SOURce <source>
```

```
TRIGger:SOURce?
```

#### Description

Sets or queries the trigger source.

**Parameter**

Name	Type	Range	Default
<source>	Discrete	{IMMEDIATE BUS EXTERNAL}	IMMEDIATE

**Description**

- The following trigger sources are available:
  - **IMMEDIATE:** immediate trigger. The trigger signal is always present. When the instrument is in the "wait-for-trigger" state, the trigger is issued immediately.
  - **BUS:** Single (Bus) trigger. When the instrument is in the "wait-for-trigger" state, you can use *\*TRG* to trigger the instrument.
  - **EXTERNAL:** external trigger. The instrument accepts the external trigger signal via the rear-panel **[EXT TRIG]** interface.
- You can use *INITiate[:IMMEDIATE]* or *READ?* to place the instrument in the "wait-for-trigger" state. A trigger will not be accepted from the selected trigger source until the instrument is in the "wait-for-trigger" state.

**Return Format**

The query returns IMM, BUS, or EXT.

**Examples**

```
TRIGger:SOURce BUS /*Sets the trigger source to Single.*/
TRIGger:SOURce? /*Queries the trigger source. The query returns
BUS.*/
```

## 4 Programming Examples

---

This chapter illustrates how to control the instrument by programming in LabVIEW, Visual Basic, and Visual C++. These examples are programmed based on Virtual Instrument Software Architecture (VISA) library.

RIGOL official website (<http://www.rigol.com>) provides the programming examples based on LabVIEWk, Visual Basic, Visual C++, Matlab, and Python. You can go to the official website and click **SUPPORT** > **Programming Demo** to obtain the programming examples.

### 4.1 Programming Preparations

---

Before programming, you need to prepare the following tasks:

You can log in to the RIGOL official website (<http://www.rigol.com>) to download the software. Then install the software according to the installation wizard. After Ultra Sigma is installed successfully, NI-VISA library will be completely installed automatically. In this manual, the default installation path is C:\Program Files\IVI Foundation\VISA.

In the manual, the instrument communicates with the PC via the USB interface. Connect the USB Device interface on the rear panel of the instrument to the PC by using the USB cable. After the instrument is properly connected to the PC, power on the instrument to start it.

The following parts will make a detailed introduction about the programming examples in LabVIEW, Visual Basic, and Visual C++.

### 4.2 LabVIEW Programming Example

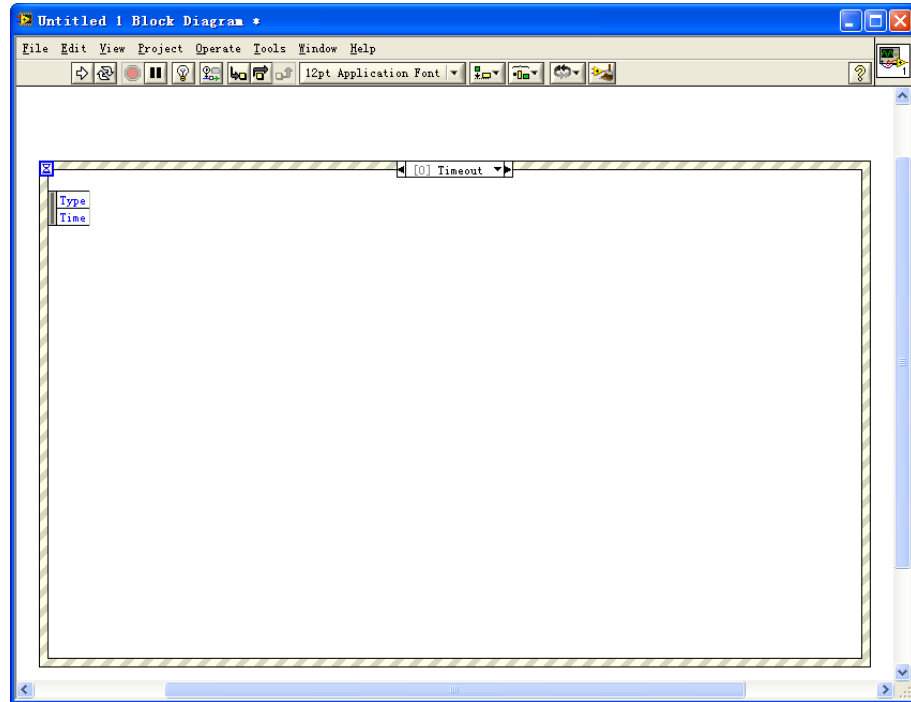
---

**Program used in this example:** LabVIEW8.6

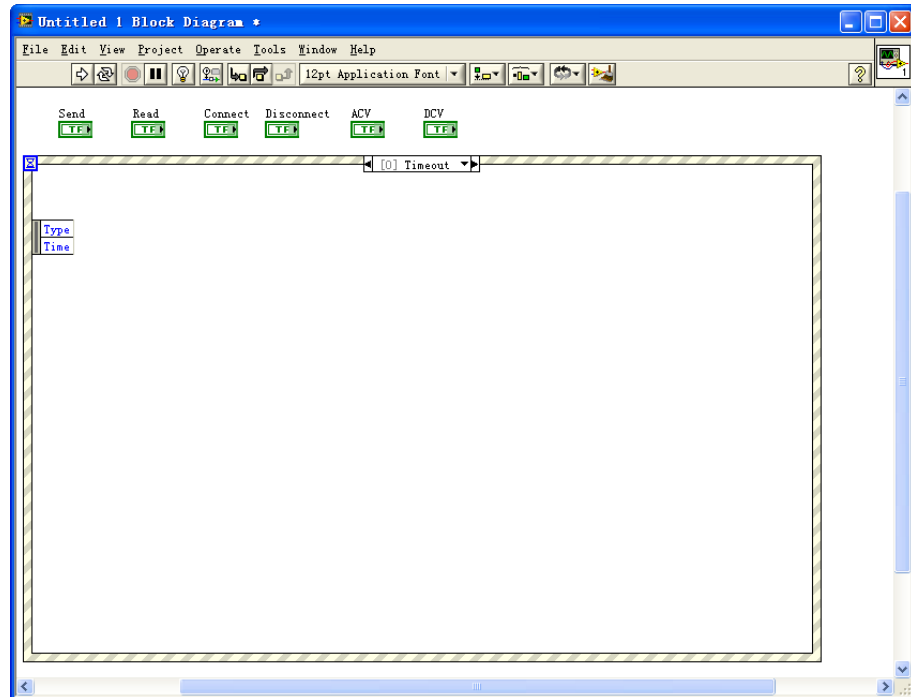
**Function realized in this example:** search for the instrument address, connect the instrument, send command, and read the returned value.

1. Create event structure

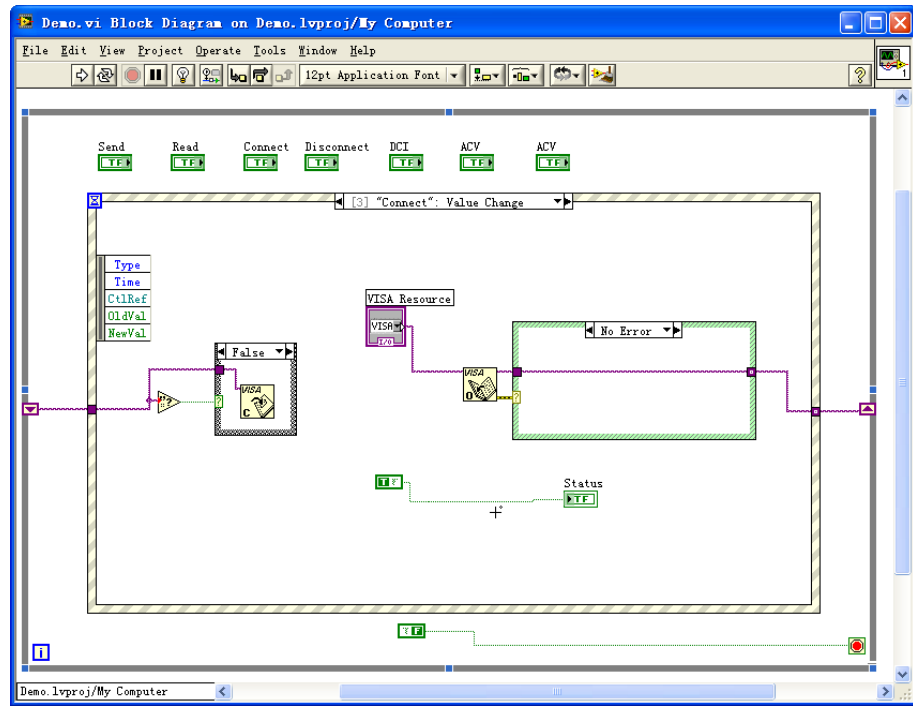




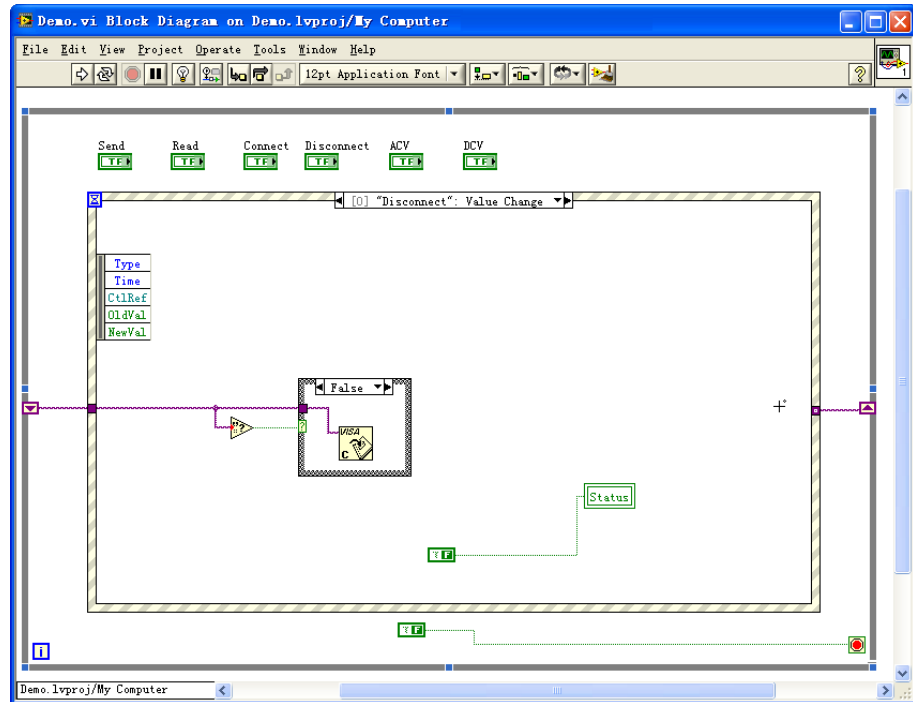
## 2. Add events



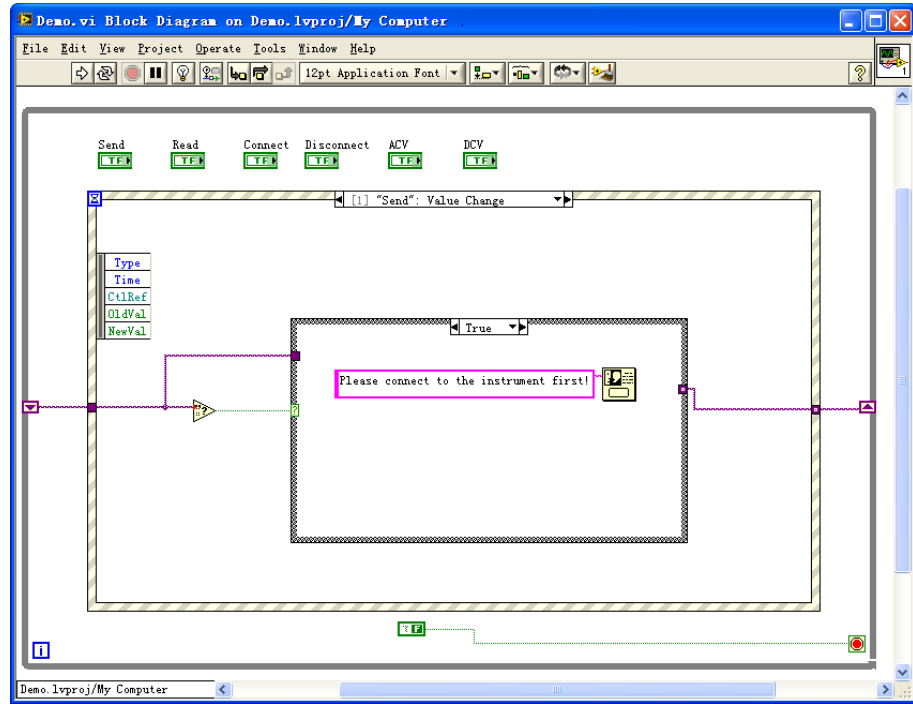
## 3. Connect the instrument code



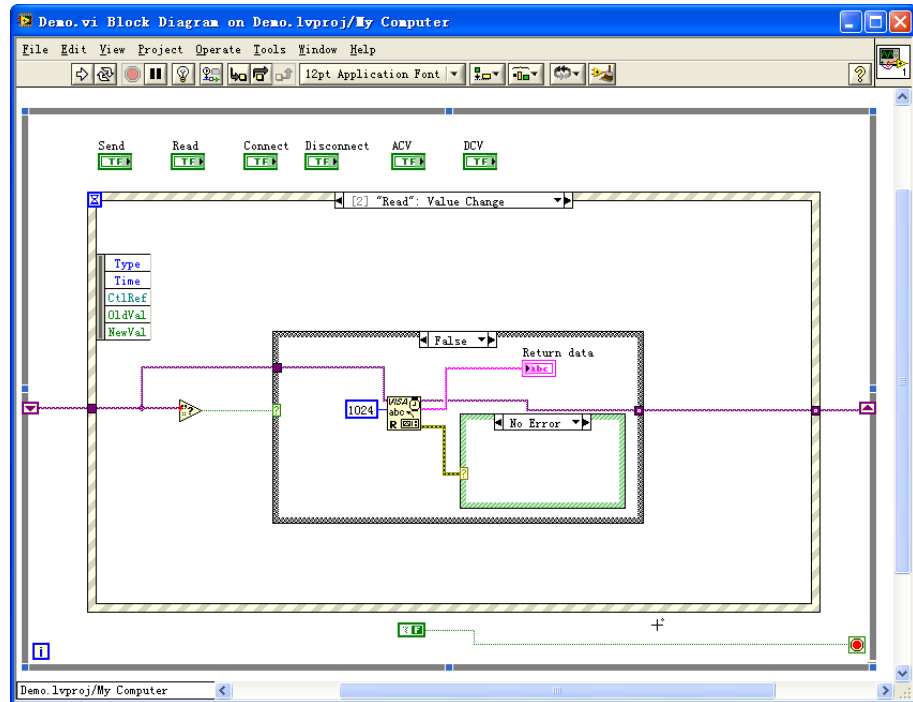
#### 4. Disconnect



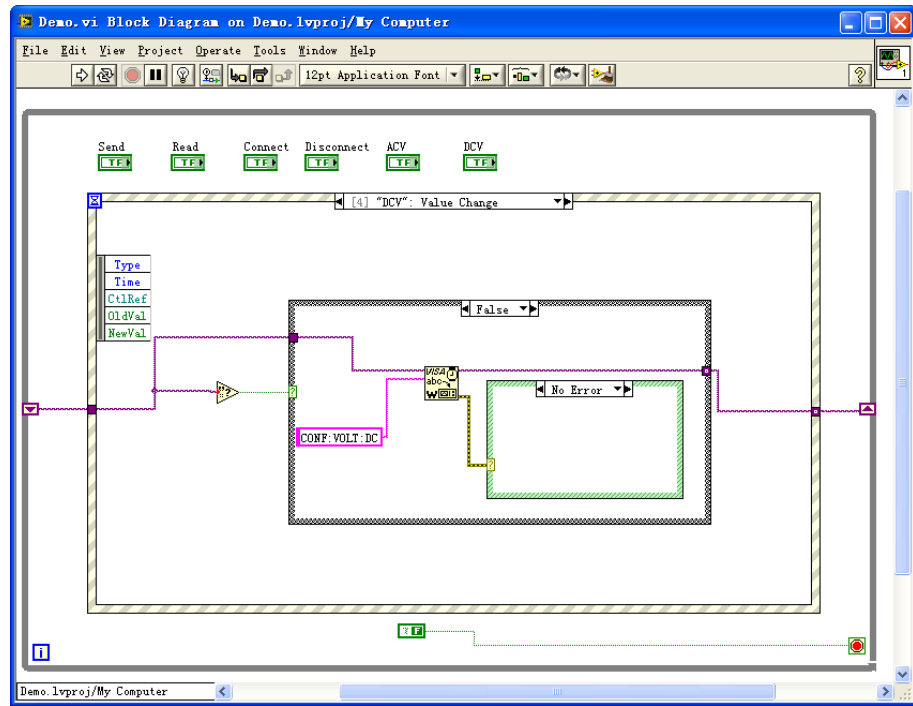
#### 5. Send the operation



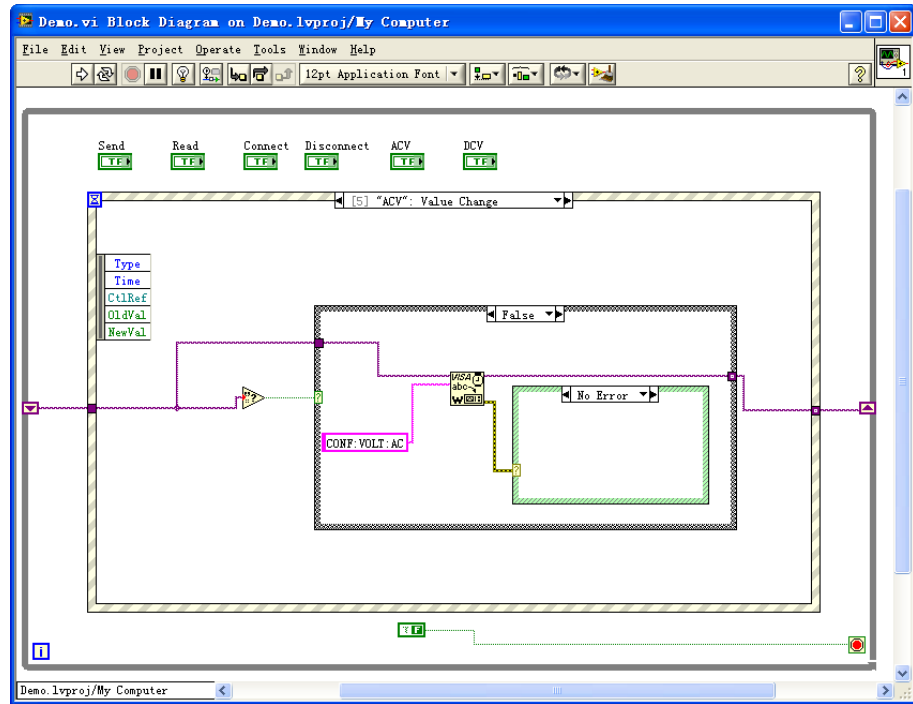
6. Read the operation



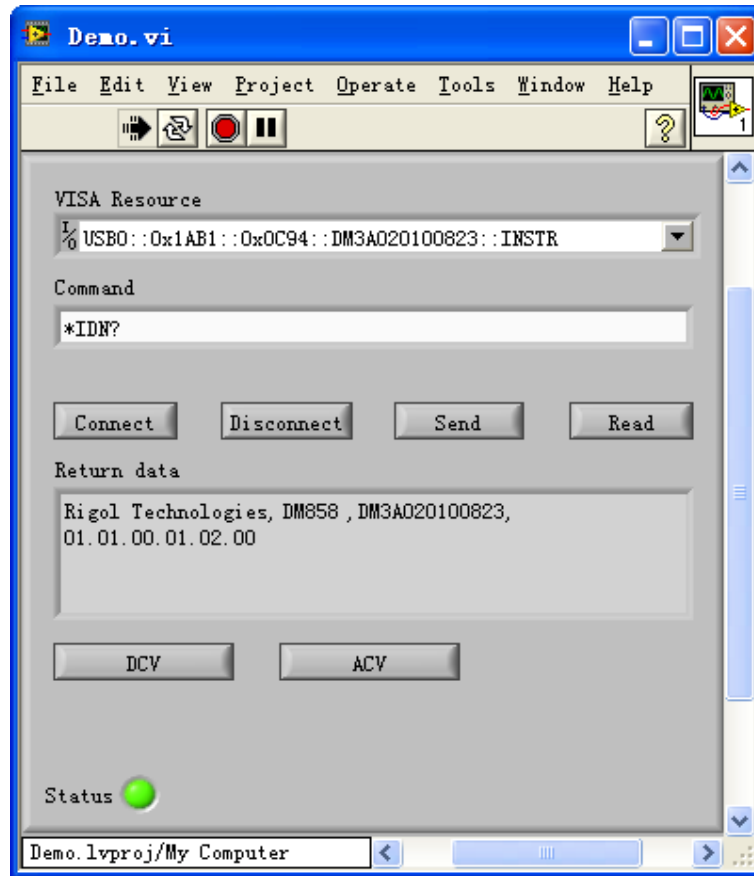
7. Switch to the DCV measurement function



8. Switch to the ACV measurement function



9. Results



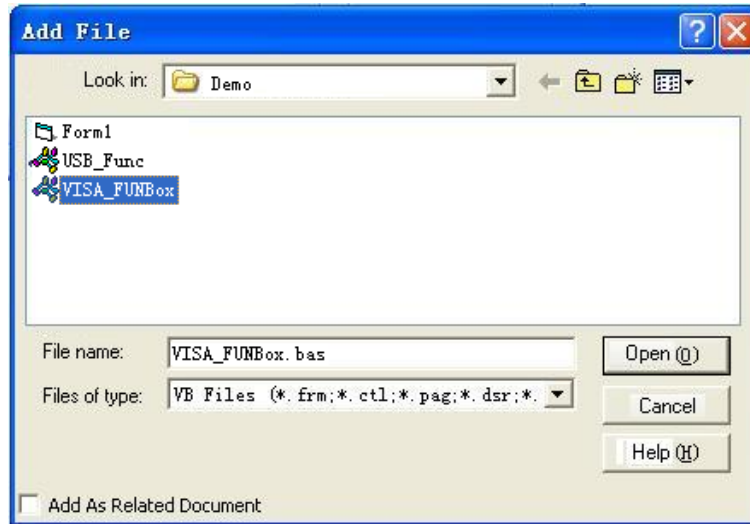
## 4.3 Visual Basic Programming Example

**Program used in this example:** Visual Basic 6.0

**Function realized in this example:** search for the instrument address, connect the instrument, send command, and read the returned value.

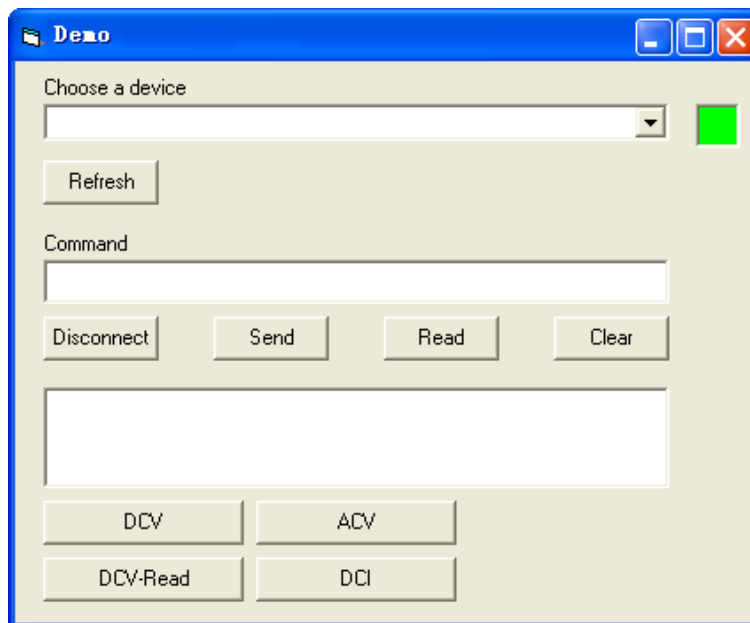
Enter the Visual Basic 6.0 programming environment, and perform the following procedures.

1. Build a standard application program project (Standard EXE), and name it "Demo".
2. Click **Project** > **Add File** to open the Add File dialog box. Add USB\_Func.bas and VISA\_FUNBox.bas to the program. The two modules include all functions and constant declarations from the VISA library used.



Then add Declare Sub Sleep Lib "kernel32" (ByVal dwMilliseconds As Long) to USB\_Func.bas. You can also create a new module to declare the Sleep function.

3. Add Text, Edit, and Button controls in Demo. The layout is as shown in the figure below.



4. Perform read and write operations for VISA encapsulation.

**a. Write operation**

```
Public Function Device_Send(cmd As String) Dim err As Long Dim cmdstr As String cmdstr = cmd + Chr$(10) err = viVPrintf(g_lVisaIO, cmdstr, 0) If (err < VI_SUCCESS) Then If CurrentLang = English Then MsgBox "Cann't Connect to Device", 48, "System Message" Else MsgBox "Device unconnected", 48, "Message" End If End If End Function
```

**b. Read operation**

```
Public Function ReadDeviceData() As String Dim strTemp As String * 256 Dim strReturn As String Dim strZero As String Dim intNpos As Integer Call viVScanf(q_lVisaIO, "%t", strTemp)
```

```
strZero = Chr(0) intNpos = InStr(strTemp, strZero) strReturn =
Left(strTemp, intNpos - 1) ReadDeviceData = strReturn End
Function
```

## 5. Add control event codes

### a. Select the instrument to connect

```
Private Sub CobDevName_Click() g_strCurrentDevice =
CobDevName.Text Dim intFuncReturn As Long Dim strReturn As
String Dim strTemp As String viClose (g_lVisaIO) intFuncReturn
= viOpen(g_lrsrcManager, g_strCurrentDevice, 0, 5000,
g_lVisaIO) If intFuncReturn >= 0 Then Device_Send ("*idn?")
strReturn = ReadDeviceData strTemp = UCase(Left(strReturn,
25)) If strTemp = "RIGOL TECHNOLOGIES,DM858" Then With
Form1 .Picture1.BackColor = &HFF00& CmdSend.Enabled = True
CmdClose.Enabled = True CmdRead.Enabled = True CmdCLR.Enabled
= True CmdDCV.Enabled = True CmdDCI.Enabled = True
CmdDCVRd.Enabled = True Cmd2NDCLR.Enabled = True End With Else
viClose (g_lVisaIO) With Form1 .Picture1.BackColor = &HFF& End
With MsgBox "Not RIGOL DM858!", 48, "Message" CmdSend.Enabled
= False CmdClose.Enabled = False CmdRead.Enabled = False
CmdCLR.Enabled = False CmdDCV.Enabled = False CmdDCI.Enabled =
False Cmd2NDCLR.Enabled = False End If Else With
Form1 .Picture1.BackColor = &HFF& End With End If End Sub
```

### b. Refresh the connection

```
Private Sub CmdRefresh_Click() With Form1 .CobDevName.Clear
End With Device_Init End Sub
```

### c. Disconnect

```
Private Sub CmdClose_Click() If Form1.Picture1.BackColor =
&HFF& Then MsgBox "Device unconnected!", 48, "Message" ' Else
Dim intFuncReturn As Integer intFuncReturn =
viClose(g_lVisaIO) If intFuncReturn >= 0 Then With
Form1 .Picture1.BackColor = &HFF& End With Else MsgBox "Device
unconnected!", 48, "Message" End If End If End Sub
```

### d. Send the command

```
Private Sub CmdSend_Click() Dim strCmd As String strCmd =
TextCmd.Text Device_Send (strCmd) End Sub
```

### e. Read the returned value

```
Private Sub CmdRead_Click() TextReturn.Text = ReadDeviceData
End Sub
```

### f. Clear the returned value

```
Private Sub CmdCLR_Click() TextReturn.Text = "" End Sub
```

### g. Switch to the DCV measurement function

```
Private Sub CmdDCV_Click() Device_Send
("configure:voltage:dc") End Sub
```

### h. Switch to the DCV measurement function and return the current reading

```
Private Sub CmdDCV_Click() Device_Send ("measure:voltage:dc?")
TextReturn.Text = ReadDeviceData End Sub
```

### i. Switch to the ACV measurement function

```
Private Sub CmdACV_Click() Device_Send
("configure:voltage:ac") End Sub
```

**j.** Switch to the DCI measurement function

```
Private Sub CmdDCI_Click() Device_Send
("configure:current:dc") End Sub
```

## 6. Results

- a. Click the "Choose a device" drop-down button to select the multimeter resource.
- b. Input "\*IDN?" into the "Command" input field.
- c. Click "Send" to write the command to the multimeter.
- d. Click "Read" to read the returned value.



## 4.4 Visual C++ Programming Example

**Program used in this example:** Visual C++6.0

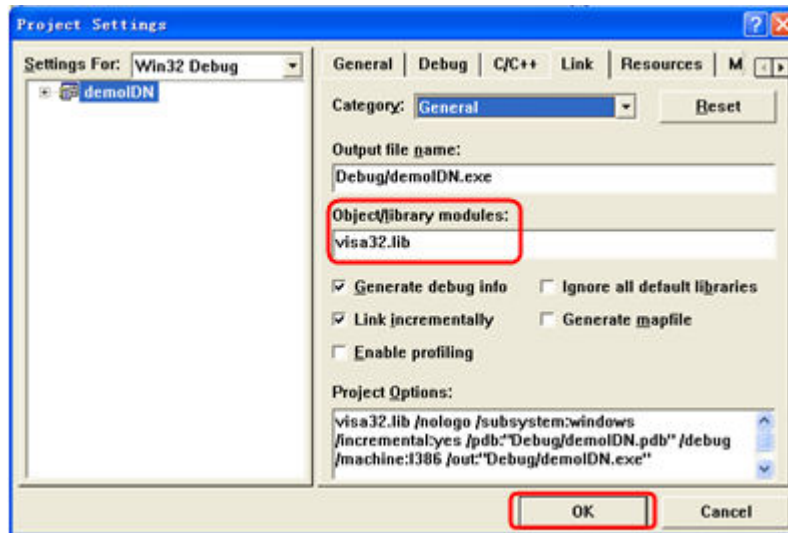
**Function realized in this example:** search for the instrument address, connect the instrument, send command, and read the returned value.

Enter the Visual C++6.0 programming environment, and perform the following procedures.

1. Create a MFC project based on a dialog box.



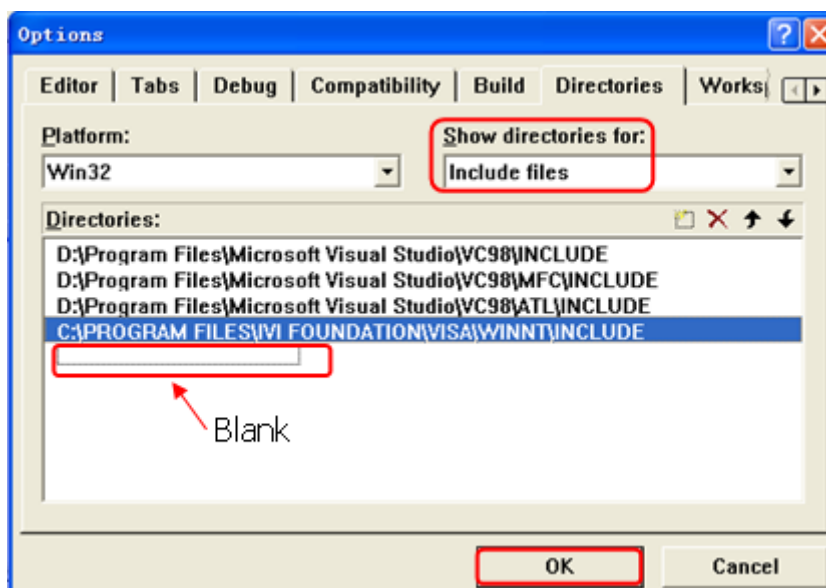
- Click Project > Settings to open the "Project Setting" dialog box. In the dialog box, click the Link tab. Add visa32.lib to the "Object/library" modules.



- Click Tools > Options to open the "Options dialog" box. Then click the Directories tab.

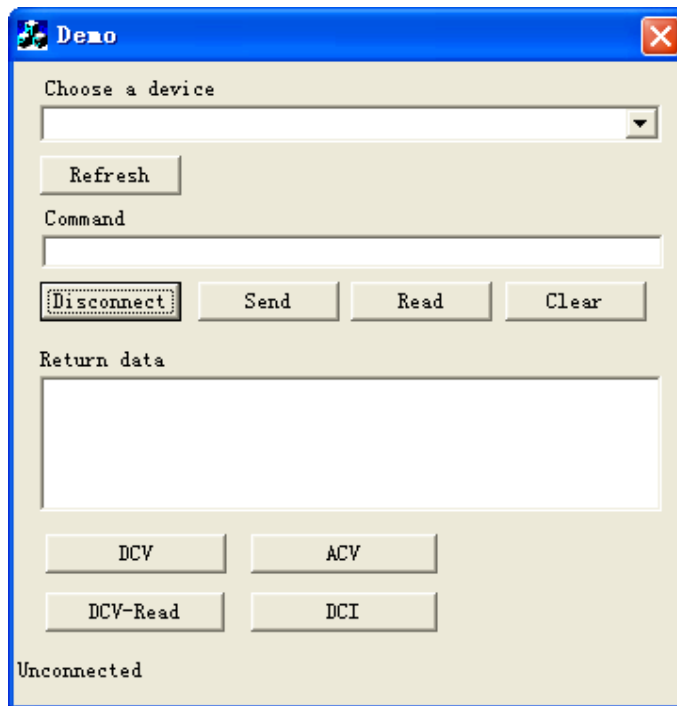
Select "Include files" from the drop-down list under "Show directories for". Double click the empty space under "Directories" to enter the specified path of Include files: C:\Program Files\IVI Foundation\VISA\WinNT\include.

Select "Library files" from the drop-down list under "Show directories for". Double click the empty space under "Directories" to enter the specified path of Library files: C:\Program Files\IVI Foundation\VISA\WinNT\lib\msc.



Note: By now, VISA library has been added.

4. Add the Text, Button, and Edit controls. The layout is as shown in the figure below.



5. Add control variables.

Open View→ClassWizard and select the Member Variables tab to add the following variables:

- Instrument ID selection box: IDC\_CBResourceName
- Instrument ID: IDC\_CBResourceName
- Refresh: IDC\_REFRESH
- Command: IDC\_CMD\_TEXT
- Disconnect: ID\_CLOSE
- Send: ID\_SEND
- Read: IDC\_READ
- Clear: IDC\_CLR
- Return data: IDC\_RETURN\_TEXT
- DCV function: IDC\_DCV
- Read the DCV measurement value: IDC\_DCVRd
- ACV function: IDC\_ACV
- DCI function: IDC\_DCI

- Connection state: IDC\_STATE\_TEXT

## 6. Perform read and write operations for VISA encapsulation.

### a. Write operation

```
//Write command to instrument void
CDemoDlg::sendToDevice(CString strCmd) { int count=0; count =
strlen(strCmd); char *charCmd = new char[count+2]; for(int i =
0;i <count;i++) { charCmd[i]=strCmd[i]; } charCmd[count]='\n';
charCmd[count+1]='\0'; unsigned long error = 0; error =
viVPrintf(g_VisaIO,charCmd,0); if(error < 0)
{ MessageBox("Communication has been cut off!!"); } else {} }
```

### b. Read operation

```
//Read data from instrument CString CDemoDlg::ReadFromDEVICE()
{ CString strReturn=""; char tempstr[256];
viScanf(g_VisaIO,"%t",tempstr); strReturn = tempstr; return
strReturn; }
```

## 7. Add control message response codes.

### a. Select the instrument to connect

```
//Connect the instrument void
CDemoDlg::OnSelchangeCBResourceName() { // TODO: Add your
control notification handler code here updateData(true); int
intIndex; intIndex = m_ResourceName.GetCurSel();
m_ResourceName.GetLBText(intIndex,g_RsrcName); unsigned long
lFuncReturn = 0; CString cstrIDNReturn = ""; char* charTemp;
charTemp = new char[26]; if(m_StateDisplay == "Connected")
{ viClose(g_VisaIO); } else {} lFuncReturn =
viOpen(g_rsrcManager,g_RsrcName,0,0,&g_VisaIO);
m_valueResourceName = g_RsrcName; if(lFuncReturn >= 0)
{ sendToDevice("*idn?"); cstrIDNReturn = ReadFromDEVICE();
strncpy(charTemp,cstrIDNReturn,25); charTemp[25]='\0';
charTemp=strupr(charTemp); CString cstrTemp=charTemp;
if(cstrTemp == "RIGOL TECHNOLOGIES,DM858") { m_StateDisplay =
"Connected"; m_btclose.EnableWindow(true);
m_btsend.EnableWindow(true); m_btread.EnableWindow(true);
m_btclr.EnableWindow(true); m_btdcv.EnableWindow(true);
m_btddci.EnableWindow(true); m_btddcvrd.EnableWindow(true);
m_bt2ndclr.EnableWindow(true); } else { viClose(g_VisaIO);
m_StateDisplay = "Unconnected"; MessageBox(" RIGOL DM858!");
m_btclose.EnableWindow(false); m_btsend.EnableWindow(false);
m_btread.EnableWindow(false); m_btclr.EnableWindow(false);
m_btdcv.EnableWindow(false); m_btddci.EnableWindow(false);
m_btddcvrd.EnableWindow(false);
m_bt2ndclr.EnableWindow(false); } UpdateData(false); } else
{ m_StateDisplay = "Unconnected"; } }
```

### b. Refresh the connection

```
//Refresh the connection void CDemoDlg::OnRefresh() { // TODO:
Add your control notification handler code here int
intItemcount = 0; intItemcount=m_ResourceName.GetCount();
for(int i=0; i < intItemcount; i++)
m_ResourceName.DeleteString(0); UpdateData(false);
m_valueResourceName = ""; ConnectToDevice(); }
```

### c. Disconnect

```
//Disconnect the connection void CDemoDlg::OnClose()
{ CDialog::OnClose(); // TODO: Add your message handler code
here and/or call default int intFuncReturn = 0;
UpdateData(true); if(m_StateDisplay == "Connected")
{ intFuncReturn = viClose(g_VisaIO); if(intFuncReturn >= 0)
{ m_StateDisplay = "Unconnected"; UpdateData(false); } else
{ MessageBox("Device unconnected!"); } } else {}
m_valueResourceName = ""; UpdateData(false); }
```

**d. Send the command**

```
//Send void CDemoDlg::OnSend() { // TODO: Add your control
notification handler code here UpdateData(true); CString cmd
=m_CmdText; sendToDevice(cmd); }
```

**e. Read the returned value**

```
//Read void CDemoDlg::OnRead() { // TODO: Add your control
notification handler code here m_ReturnText =
ReadFromDevice(); UpdateData(false); }
```

**f. Clear the returned value**

```
//Clear the data from the return data dialog void
CDemoDlg::OnClr() { // TODO: Add your control notification
handler code here m_ReturnText = ""; UpdateData(false); }
```

**g. Switch to the DCV function**

```
// Switch to DCV void CDemoDlg::OnDcv() { // TODO: Add your
control notification handler code here
sendToDevice("CONFIGURE:VOLTAGE:DC"); }
```

**h. Switch to the DCV function and return the current reading**

```
// Switch to DCV and return its reading void
CDemoDlg::OnDCVRd() { sendToDevice("MEASURE:VOLTAGE:DC?");
m_ReturnText = ReadFromDevice(); UpdateData(false); }
```

**i. Switch to the ACV function**

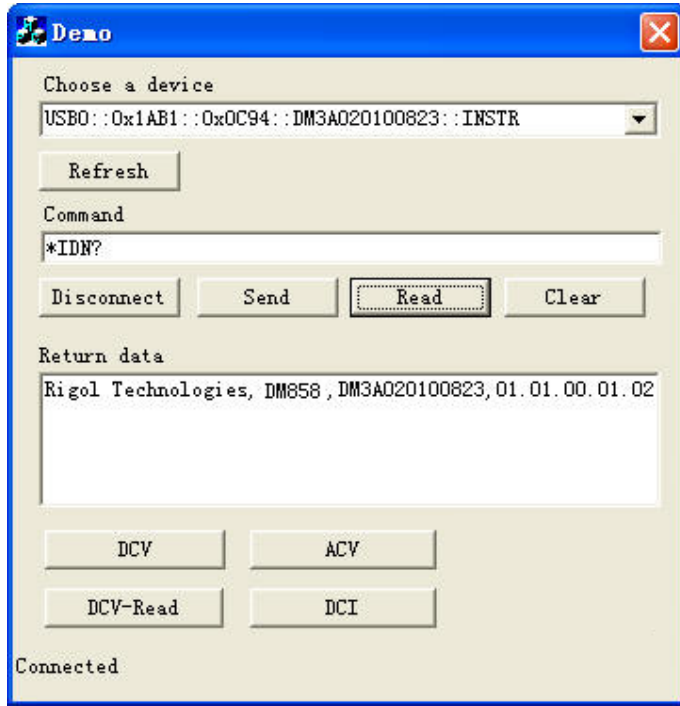
```
// Switch to ACV void CDemoDlg::OnAcv() { // TODO: Add your
control notification handler code here
sendToDevice("CONFIGURE:VOLTAGE:AC"); }
```

**j. Switch to the DCI function**

```
// Switch to DCI void CDemoDlg::OnDci() { // TODO: Add your
control notification handler code here
sendToDevice("CONFIGURE:CURRENT:DC"); }
```

**8. Results**

- a. Click the "Choose a device" drop-down button to select the multimeter resource.
- b. Input "\*IDN?" into the "Command" input field.
- c. Click "Send" to write the command to the multimeter.
- d. Click "Read" to read the returned value.



#### HEADQUARTER

**RIGOL TECHNOLOGIES CO., LTD.**  
No.8 Keling Road, New District, Suzhou,  
JiangSu, P.R.China  
Tel: +86-400620002  
Email: info@rigol.com

#### EUROPE

**RIGOL TECHNOLOGIES EU GmbH**  
Carl-Benz-Str.11  
82205 Gilching  
Germany  
Tel: +49(0)8105-27292-0  
Email: info-europe@rigol.com

#### NORTH AMERICA

**RIGOL TECHNOLOGIES, USA INC.**  
10220 SW Nimbus Ave.  
Suite K-7  
Portland, OR 97223  
Tel: +1-877-4-**RIGOL**-1  
Fax: +1-877-4-**RIGOL**-1  
Email: info@rigol.com

#### JAPAN

**RIGOL JAPAN CO., LTD.**  
5F,3-45-6,Minamiotsuka,  
Toshima-Ku,  
Tokyo,170-0005,Japan  
Tel: +81-3-6262-8932  
Fax: +81-3-6262-8933  
Email: info-japan@rigol.com

---

**RIGOL**® is the trademark of **RIGOL TECHNOLOGIES CO., LTD.** Product information in this document is subject to update without notice. For the latest information about **RIGOL**'s products, applications and services, please contact local **RIGOL** channel partners or access **RIGOL** official website: [www.rigol.com](http://www.rigol.com)