Multimeter SCPI Protocol

Introduction to the SCPI Language

Syntax

SCPI commands present a hierarchical tree structure and contain multiple sub-systems, each of which is made up of a root keyword and one or more sub-keywords. The command string usually starts with ":", the keywords are separated by ":" and are followed by the parameter settings available, "?" is added at the end of the command string to indicate query and the command and parameter are separated by "space".

For example,
:VOLT:AC:AUTO {OFF|ON}

VOLT is the root keyword of the command. **AC** and **AUTO**are the second-level and third-level keywords respectively. The command string starts with "**:**" which separates the multiple-level keywords. **{OFF|ON}** represents parameters available for setting, "**?**" represents query and the command **:VOLT:AC:AUTO** and the parameter **{OFF|ON}** are separated by "space".

Syntax Rules

SCPI language itself defines a group of sub-system keywords, and at the same time allows users to add or reduce keywords. Those keywords can be some meaningful English words and are easy to remember, which are called mnemonics. Mnemonic has long and short types. The short are the abbreviation of the long.

- **Rule to format mnemonics:**

1) If the letter number of an English word is less than or equal to 4, then the word itself can be the mnemonic.(such as “Free” can be “FREE”)
2) If the letter number of an English word exceeds 4, then the first four letters will be the mnemonic.(such as “Frequency” can be “FREQ”)
3) If the forth letter is vowel, then mnemonic uses the former three letters. Vowels consists of a, e, i, o, and u.(such as “Power” can be “POW”)
4) If it is not a word but a sentence, then use the first letters of the former words and the whole of the last word. (such as “Input Voltage “ can be “IVOLtage”)

- **Usage of symbols**

1) Space

The space is used to separate command and parameter.

2) Colon :

If the colon is in front of the first character, it means the following is Root Command. When the colon is set between two keywords, then it means moving from the current level to the next level.

3) *asterisk

The commands start with asterisk are named Common Command, which is used to execute IEEE488.2 common commands.

4) Braces {}

The parameters enclosed in the braces are optional and are usually separated by the vertical bar "|". When using this command, one of the parameters must be selected.

5) Vertical Bar |

The vertical bar is used to separate multiple parameters and one of the parameters must be selected when using the command.

6) Triangle Brackets < >

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

- **Parameter Type**

  Bool

  The parameter could be "OFF", "ON". For example,

  :VOLT:AC:AUTO {OFF|ON}

  wherein,

  <bool> can be set to {OFF|ON}

Command Abbreviation

Each SCPI command can be written mixed with uppercase and lowercase according to the syntax rules, and the capital letter part is just the abbreviation of the command. If abbreviation is used, all the capital letters in the command must be written completely. For parameters with units, please refer to the detail parameter specifications in the sub-system.

Contact Us

If you have any problem or requirement when using our products, please contact OWON.

Service & Support Hotline: **4006 909 365**

Website：http://www.owon.com.cn/

The Entry of the third party of programming

Support USB connecting. Open PC software, click DMM SCPI control icon, then the software sends command :SCPI:DISP? , if the device supports SCPI, then it will send back :SCPION . After receiving that command, the device will open DMM SCPI control interface. If the device sends nothing, then a warning would pop up to tell you the device does not support SCPI protocol.

ATTENTION: Due to DMM SCPI commands are comparatively easy. Here we just use abbreviation and send control command directly. Only one query command is provided for reading data, that is :READ? .

Multimeter SCPI Commands

Select the function of the multimeter

:FUNC	DCV	set multimeter to measure DC Voltage
:FUNC	ACV	set multimeter to measure AC Voltage
:FUNC	DCA	set multimeter to measure Direct Current
:FUNC	ACA	set multimeter to measure Alternating Current
:FUNC	RES	set multimeter to measure Resistance
:FUNC	DIOD	set multimeter to measure Diode Voltage
:FUNC	BEEP	Enable the Buzzer of the multimeter
:FUNC	CAP	set multimeter to measure Capacity

Voltage Measurement

:VOLT:AC:AUTO	{OFF|ON}	ACV automatic measurements
:VOLT:DC:AUTO	{OFF|ON}	DCV automatic measurements
:VOLT:AC:REL	{ON|OFF}	ACV relative value
:VOLT:DC:REL	{ON|OFF}	DCV relative value

Measure Range

DCV	X1	-4V~4V	:VOLT:DC:RANG 4
	X10	-40V~40V	:VOLT:DC:RANG 40
	X100	-400V~400V	:VOLT:DC:RANG 400
	X1000	-1000V~1000V	:VOLT:DC:RANG 1000
	X100	-400mV~400mV	:VOLT:DC:RANG 4E-1
ACV	X1	0~4V	:VOLT:AC:RANG 4
	X10	0~40V	:VOLT:AC:RANG 40
	X100	0~400V	:VOLT:AC:RANG 400
	X1000	0~1000V	:VOLT:AC:RANG 1000

Current Measurement

- When “Manual” mode

:CURR:AC:AUTO	{OFF|ON}	ACA automatic measurements
:CURR:DC:AUTO	{OFF|ON}	DCA automatic measurements
:CURR:AC:REL	{ON|OFF}	ACA relative value
:CURR:DC:REL	{ON|OFF}	DCA relative value

Measure Range

DCA	mA		:CURR:DC:UNIT mA
	X10	-40mA~40mA	:CURR:DC:RANG 4E-2
	X100	-400mA~400mA	:CURR:DC:RANG 4E-1
DCA	10A		:CURR:DC:UNIT 10A
	X1	-4A~4A	:CURR:DC:RANG 4
	X10	-10A~10A	:CURR:DC:RANG 10
ACA	mA		:CURR:AC:UNIT mA
	X10	0mA~40mA	:CURR:AC:RANG 4E-2
	X100	0mA~400mA	:CURR:AC:RANG 4E-1
ACA	10A		:CURR:AC:UNIT 10A
	X1	0A~4A	:CURR:AC:RANG 4
	X10	0A~10A	:CURR:AC:RANG 10

Measure Resistance

:RES:AUTO {OFF|ON} Resistance automatic measurements

Measure Range

RES	Ω	:RES:RANG	OHM
	KΩ	:RES:RANG	KOHM
	MΩ	:RES:RANG	MOHM

Measure Capacity

:CAP:REL {ON|OFF}

Measure Diode Voltage

Enable Buzzer

Read Data

:READ?

Send :READ? to read the measured value of the current function. And the format of the data like this. DCV 0.300000V .

IEEE488.2 Common Commands

*CLS

Clear all the event registers in the register set and clear the error queue.

*ESE

Set enable register for the standard event register set.

Parameter

Name	Type	Range	Default Value
<value>	Integer	0 to 255	0

Explanation

The bit 1 and bit 6 of the standard event register are not used and are always treated as 0, therefore, the range of <value> are the decimal numbers corresponding to the binary numbers ranging from 00000000 (0 in decimal) to 11111111 (255 in decimal) and of which the bit 1 and bit 6 are 0.

Definitions of the Bits in ESE Register:

Bit	weights	Name	Enable
7	128	PON	Power On
6 (Not used)	64	URQ	User Request
5	32	CME	Command Error
4	16	EXE	Execution Error
3	8	DDE	Dev. Dependent Error
2	4	QYE	Query Error
1 (Not used)	2	RQL	Request Control
0	1	OPC	Operation Complete

Return Format

The query returns an integer which equals to the sum of the weights of all the bits that have already been set in the register. For example, the query returns "144" if bit 4 (16 in decimal) and 7 (128 in decimal) are enabled.

Example

The command below enables bit 4 (16 in decimal) of the enable register.

*ESE 16

The query below returns "16".

*ESE?

*ESE?

Query which bit in ESE register is enabled.

Example

The command below enables bit 4 (16 in decimal) of the enable register.

*ESE 16

The query below returns "16".

*ESE?

*ESR?

Description

Query the event register for the standard event register set.

Parameter

Name	Type	Range	Default Value
<value>	Integer	0 to 255	0

Explanation

The bit 1 and bit 6 of the standard event register are not used and are always treated as 0, therefore, the query returns the decimal numbers corresponding to the binary numbers ranging from 00000000 (0 in decimal) to 11111111 (255 in decimal) and of which the bit 1 and bit 6 are 0.

Definitions of the Bits in ESE Register:

Bit	weights	Name	Enable
7	128	PON	Power On
6 (Not used)	64	URQ	User Request
5	32	CME	Command Error
4	16	EXE	Execution Error
3	8	DDE	Dev. Dependent Error
2	4	QYE	Query Error
1 (Not used)	2	RQL	Request Control
0	1	OPC	Operation Complete

Return Format

The query returns an integer which equals to the sum of the weights of all the bits that have already been set in the register. For example, the query returns "144" if bit 4 (16 in decimal) and 7 (128 in decimal) are enabled.

Example

The query below returns "24" (bit 3 and bit 4 have already been set).

*ESR?

*IDN

Return the ID character string of the instrument.

Description

The query returns the ID character string of the instrument.

Return Format

OWON,<model>,<serial number>,X.XX.XX

<model>：the model number of the instrument.

<serial number>：the serial number of the instrument.

X.XX.XX：the software version of the instrument.

Example

OWON,SDS6062,1247048,v3.0.2

*OPC

Set the "Operation Complete" bit in the standard event register to 1 after the current operation is finished.

*OPC?

Query whether the current operation is finished.

Explanation

Note the difference between the *OPC? and *OPC commands: the latter sets the "Operation Complete" bit (bit 0) in the standard event register to 1 after the current operation is finished.

Return Format

The query returns "1" if the current operation is finished, otherwise returns "0".

*RST

Restore the instrument to its default value.

*SRE

Set enable register for the state byte register set.

Parameter

Name	Type	Range	Default Value
<value>	Integer	0 to 255	0

Explanation

The bit 0 and bit 1 of the state byte register are not used and are always treated as 0, therefore, the range of <value> are the decimal numbers corresponding to the binary numbers ranging from 00000000 (0 in decimal) to 11111111 (255 in decimal) and of which the bit 0 and bit 1 are 0.

Definitions of the Bits in SRE:

Bit	Weights	Name	Enable
7	128	OPER	Operation Status Reg
6	64	---	Not used
5	32	ESB	Event Status Bit
4	16	MAV	Message Available
3	8	---	Not used
2	4	MSG	Message
1 (Not used)	2	USR	User
0 (Not used)	1	TRG	Trigger

Return Format

The query returns an integer which equals to the sum of the weights of all the bits that have already been set in the register. For example, the query returns "144" if bit 4 (16 in decimal) and 7 (128 in decimal) are enabled.

Example

The command below enables bit 4 (16 in decimal) of the enable register.

*SRE 16

The query below returns "16".

*SRE?

*STB?

Query the condition register for the state byte register set.

*TST?

Perform self-test and return the test result.
If the returned bit is "0", the corresponding item of the instrument passed this test, while "1" indicates a failure.

*WAI

Wait for the finish of the operation.