UNI-T®

Instruments.uni-trend.com

Programming Manual

UDP3000S Series Programming DC Power Supply

Warranty and Statement

Brand Information

UNI-T is the registered trademark of Uni-Trend Technology (China) Co., Ltd.

Software Version

1.10

Software upgrade may have some change and add more function, please subscribe UNI-T website to get the new version or contact UNI-T.

Statement

- UNI-T products are protected by patents (including obtained and pending) in China and other countries and regions.
- UNI-T reserves the right to change specifications and prices.
- The information provided in this manual supersedes all previous publications.
- Information provided in this manual is subject to change without prior notice.
- UNI-T shall not be liable for any errors that may be contained in this manual. For any incidental or consequential damages, arising out of the use or the information and deductive functions provided in this manual.
- Without the written permission of UNI-T, this manual cannot photocopied, reproduced or adapted.

Product Certification

UNI-T has certified that the product conforms to China national product standard and industry product standard as well as ISO9001:2008 standard and ISO14001:2004 standard. UNI-T will go further to certificate product to meet the standard of other member of the international standards organization.

SCPI

SCPI (Standard Commands for Programmable Instruments) is a standard command set based on the existing standards IEEE 488.1 and IEEE 488.2. And follow the IEEE754 standard floating-point arithmetic rules, ISO646 information exchange 7 bits code symbol (equivalent to ASCII programming) and other standard standardized instrument programming language.

This section describes the format, symbols, parameters, and abbreviation rules of the SCPI command.

Instruction Format

The SCPI command is a tree-like hierarchy consisting of multiple subsystems, each consisting of a root keyword and one or more hierarchical key words.

The command line usually begins with a colon ":"; Keywords are separated by the colon ":", followed by optional parameter settings. The command keyword is separated by spaces from the first parameter. The command string must end with a newline <NL> character. Add the question mark "? " after the command line. It is usually indicated that this feature is being queried.

Symbol Description

The following four symbols are not part of SCPI command, it cannot send with the command. It usually used as supplementary description of command parameter.

- Braces { }
  usually contains multiple optional parameters, it should select one parameter when send command.
  For example, INSTrument[:SELEct] {CH1|CH2|CH3|SER|PARA}

- Vertical bar |
  used to separated multiple parameters, it should select one parameter when send command.
  For example, INSTrument[:SELEct] {CH1|CH2|CH3|SER|PARA}

- Square Brackets [ ]
  the contents in square brackets (command keywords) can be omissible. If the parameter is ignored, the instrument will set the parameter as the default value.
  For example, for the command : INSTrument[:SELEct] {CH1|CH2|CH3|SER|PARA}, [:SELEct] can be ignored.

- Triangular Brackets < >
  the parameter in the brackets must be replaced with a valid value.
  For example, :LISTout:TEMPlet:POINTs <point>
  :LISTout:TEMPlet:POINTs 20

Parameter Description

The parameter in this manual can divide into five types: Boolean, Integer, Real, Discrete and ASCII string.

- Boolean
  Parameter value can set "ON" (1) or "OFF" (0).

For example, :LISTout[:STATe] {0|1|OFF|ON}

- **Integer**
  Unless otherwise specified, the parameter can take any valid integer value.
  Note: Do not set decimal as parameter, otherwise it may occur error.

- **Real**
  Unless otherwise specified, the parameter can take any valid integer value.

- **Discrete**
  Parameter can only take some specified numbers or characters.
  For example, the parameter in the command :SOURce:Mode {NORMal|SER|PARA} can only be NORMal, SER or PARA.

- **ASCII String**
  String parameter contain all ASCII string sets. Strings must begin and end with paired quotes; it can use single or double quotation marks. The quotation and delimiter can also be part of a string by typing it twice and not adding any characters.
  For example, set IP: SYST:COMM:LAN:IPAD "192.168.1.10"

Shorthand Rule

All commands are case sensitivity. If command writes in abbreviation format, all capital letters in the command should input completely.

Data Return

Data return is divided into single data and batch data. The single data return is the corresponding parameter type, in which the real return type is express in scientific notation. The part before e retains three figure behind the decimal point, and the e part retains three figure; the batch return must be obey IEEE 488.2# string data format, '#'+ the length of character bits [fixed to one character] + ASCII valid value+ valid data+ end mark ['\n']

For example, #3123xxxxxxxxxxxxxxxxxxx\n represents 123 strings batch data return format, '3' presents "123" occupies three character bits.

Note: If return data is invalid data, use * to represent it.

Communication

The instrument can communicate with the computer via USB, LAN and RS232 interface. The end mark is "\n", that is hexadecimal system "0x0A".

USB Interface

The instrument as a USB-TMC device. VISA programming resource descriptor is similar as:

"USB0::0x0483::0x5740::UDP51183557335E::INSTR"

LAN Interface

The instrument supports the programming mode of VXI and SOCKET. The port number of SOCKET mode is 5025. VISA programming resource descriptor is similar as:

"TCPIP0::192.168.10.142::INSTR"

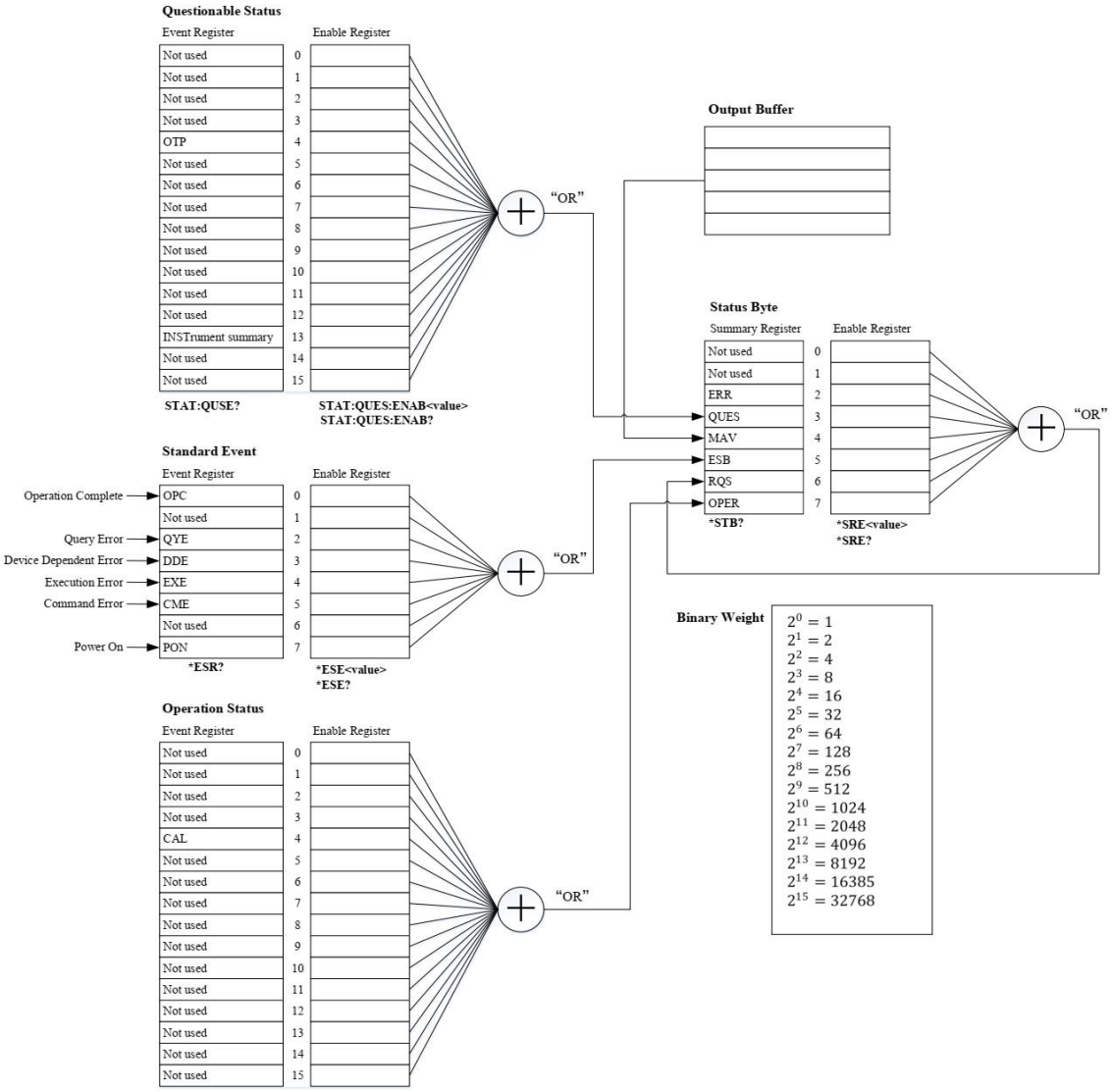
"TCPIP0::192.168.10.142::5025::SOCKET"

RS232 Interface

Please use cross RS232 connecting line and select the baud rate according to the actual situation. The instrument supports 4800, 7200, 9600, 14400, 19200, 38400, 57600, 115200, 128000. It can set in the instrument's system.

SCPI Status Register

Standard Register



The definition of suspicious status register

Bit	Definition	Decimal	Meaning
4	OTP	16	Over temperature protection
13	INSTrument summary	8192	The summary of the channel's suspicious status register and the channel's SUMMARY register
Others	Not used	-	Not used, always be 0

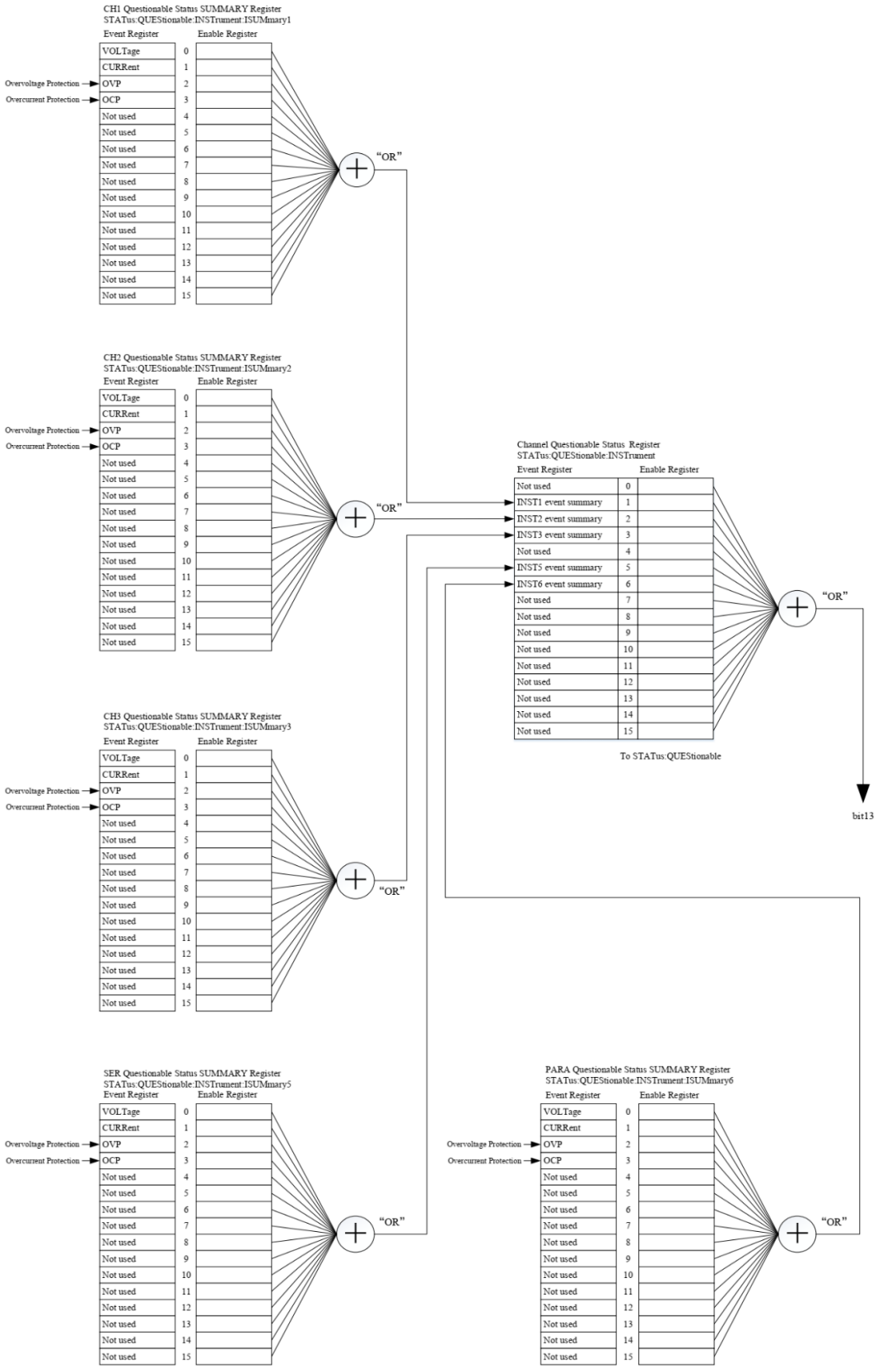
The definition of standard event register

Bit	Definition	Decimal	Meaning
0	OPC	1	The operation is completed. *OPC instruction is executed.
1	Not used	2	Not used, always be 0
2	QYE	4	Query error
3	DDE	8	Device error
4	EXE	16	Execution error
5	CME	32	Command error
6	Not used	64	Not used, always be 0
7	PON	128	Power-on inspection

The definition of stauts bit

Bit	Definition	Decimal	Meaning
0	Not used	1	Not used, always be 0
1	Not used	2	Not used, always be 0
2	ERR	4	Message contains in error output queue
3	QUES	8	Suspicious status register reports the event
4	MAV	16	The data in output buffer can be used
5	ESB	32	Standard event register reports the event
6	RSQ	64	Request service
7	OPER	128	Operation register reports the event

Channel's suspicious status register



The definition of the channel's SUMMARY register

Bit	Definition	Decimal	Meaning
0	VOLTage	1	The voltage is not controllable. The power is in the constant current mode.
1	CURRent	2	The current is not controllable. The power is in the constant voltage mode.
2	OVP	4	Over voltage protection
3	OCP	8	Over current protection
Others	Not used	-	Not used, always be 0

The definition of the channel's suspicious status register

Bit	Definition	Decimal	Meaning
0	Not used	1	Not used, always be 0
1	INST1 event summary	2	The event summary of CH1 independent channel
2	INST2 event summary	4	The event summary of CH2 independent channel
3	INST3 event summary	8	The event summary of CH3 independent channel
4	Not used	16	Not used, always be 0
5	INST5 event summary	32	The event summary of SER channel
6	INST6 event summary	64	The event summary of PARA channel
Others	Not used	-	Not used, always be 0

SCPI Command Explanation

1. IEEE488 Common Command

*CLS

Format: *CLS

Function: Set the value of the enable register in standard event register.

*ESE

Format: *ESE <enable value>

Function: Set the value of the enable register in standard event register.

*ESE?

Format: *ESE?

Function: Query the value of the enable register in standard event register.

*ESR?

Format: *ESR?

Function: Query the value of the enable register in standard event register.

*IDN?

Format: *IDN?

Function: Query the basic information of the instrument, such as the manufacture's name, model, product serial number and software version.

*OPC

Format: *OPC

Function: "OPC"bit is set in standard event register after this command is executed.

*OPC?

Format: *OPC?

Function: Return"1"when all commands are executed.

*RST

Format: *RST

Function: Restore the instrument to the factory settings.

*SRE

Format: *SRE <enable value>

Function: Set the value of the enable register in status register.

***SRE?**

Format: *SRE?

Function: Query the value of the enable register in status register.

***STB?**

Format: *STB?

Function: Query the status byte register.

***TST?**

Format: *TST?

Function: Query the result of the instrument's self-inspection. It returns 0 if the instrument is normal, it returns 1 if the instrument is abnormal.

***WAI**

Format: *WAI

Function: Set the instrument to wait to complete the unfinish operation and then to execute other commands.

***SAV**

Format: *SAV{1|2|3|4|5|6|7|8|9|10}

Function: Save the status data to the specified position of the internal memory.

Description: It equals to the command":MEMory[:STATe]:STORe STA, {1|2|3|4|5|6|7|8|9|10}"

For Example: *SAV, 1

***RCL**

Format: *RCL{1|2|3|4|5|6|7|8|9|10}

Function: Loading the status data of the specified position from the internal memory.

Description: It equals to the command":MEMory[:STATe]:LOAD STA, {1|2|3|4|5|6|7|8|9|10}"

For Example: *RCL, 1

2. APPLy

:APPLy

Format: **:APPLy** [CH1|CH2|CH3|SER|PARA],[<volt>|MINimum|MAXimum], [<curr>|MINimum|MAXimum]

Function: Set the specified channel as the current channel, and set the voltage and current value for this channel.

Description: "CH1, CH2"can only be selected in normal mode; "SER"can only be selected in serial mode; "PARA"can only be selected in parallel mode. When"volt"and "curr"parameter is ignored at the same time, it only sets the the specified channel as the current channel, not change the voltage and current value of the channel.

For Example: :APPLy CH1,15.00V, 2.000A

:APPLy?

Format: **:APPLy?** [CH1|CH2|CH3|SER|PARA],[CURRent|VOLTage]

Function: Query the voltage and current value of a certain channel.

Description: If the channel parameter is ignored, then the current channel is the channel which need to be queried by default. If it assigns "CURRent", then return the current value; If it assigns "VOLTage", then return the voltage value; If it has no specified assignment, then return the voltage and current value.

For Example: :APPLy? CH1, VOLT

Return: CH1, 15.00

3. INSTrument

:INSTrument[:SELEct]

:INSTrument[:SELect]

Format: **:INSTrument[:SELect]**{CH1|CH2|CH3|SER|PARA}

:INSTrument[:SELEct]{CH1|CH2|CH3|SER|PARA}

Function: Set the specified channel as the current channel.

Description: "CH1, CH2"can only be selected in normal mode; "SER"can only be selected in serial mode; "PARA"can only be selected in parallel mode.

Set the power mode refer to the command ":SOURce:Mode {NORMal|SER|PARA}"

For Example: :INSTrument:SELE Ch3

:INSTrument[:SELEct]?

:INSTrument[:SELect]?

Format: **:INSTrument[:SELEct]?**

:INSTrument[:SELect]?

Function: Query the current channel.

Description: Return"CH1|CH2|CH3|SER|PARA".

For Example: :INSTrument:SELE?

Return: CH1

:INSTrument:NSELect

Format： **:INSTrument:NSELect** {1|2|3|5|6}

Function： Set the specified channel as the current channel.

Description： 1 is corresponding to CH1; 2 is corresponding to CH2; 3 is corresponding to CH3; 5 is corresponding to SER; 6 is corresponding to PARA. "1, 2"can only be selected in normal mode; "5" can only be selected in serial mode; "6"can only be selected in parallel mode.

For Example： :INSTrument:NSELect 3

:INSTrument:NSELect?

Format： **:INSTrument:NSELect?**

Function： Query the current channel.

Description： Return"1|2|3|5|6".

For Example： :INSTrument:NSELect?

Return：3

4. SOURce

:SOURce:Mode

Format： **:SOURce:Mode** {NORMal|SER|PARA}

Function： Set the work mode of the power supply.

Description： NORMal is the normal independent mode; SER is the serial mode; PARA is the parallel mode. It takes some time for the power supply to switch modes, during this time if executing the commands related to work mode of the power supply, it may cause the command execution to fail. Therefore, after switching work mode of the power supply, a new command is executed after an interval of at least 500 milliseconds.

For Example： :SOURce:Mode SER

:SOURce:Mode?

Format： **:SOURce:Mode?**

Function： Query the current work mode of the power supply.

Description： Return"NORMAL| SER|PARA".

For Example： :SOURce:Mode?

Return：SER

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

Format： **[:SOURce#]:VOLTage[:LEVel][:IMMediate][:AMPLitude]** {<vol>|MINimum|MAXimum}

Function： Set the voltage value of a certain channel and set the channel as the current channel.

Description: The optional value for # are"1|2|3|5|6". 1 is corresponding to CH1; 2 is corresponding to CH2; 3 is corresponding to CH3; 5 is corresponding to SER; 6 is corresponding to PARA. If [:SOURce#] or # is ignored, then the default is CH1. "1, 2"can only be selected in normal mode; "5" can only be selected in serial mode; "6"can only be selected in parallel mode.

For Example: :SOURce1:VOLTage 25.00

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

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

Function: Query the voltage value of a certain channel.

Description: The optional value for # are"1|2|3|5|6". 1 is corresponding to CH1; 2 is corresponding to CH2; 3 is corresponding to CH3; 5 is corresponding to SER; 6 is corresponding to PARA. If [:SOURce#] or # is ignored, then the default is CH1.

For Example: :SOURce1:VOLTage?

Return: 25.00

[:SOURce#]:VOLTage:PROTection[:LEVel]

Format: [:SOURce#]:VOLTage:PROTection[:LEVel] {<vol>|MINimum|MAXimum}

Function: Set the overvoltage protective value of a certain channel and set the channel as the current channel.

Description: The optional value for # are"1|2|3|5|6". 1 is corresponding to CH1; 2 is corresponding to CH2; 3 is corresponding to CH3; 5 is corresponding to SER; 6 is corresponding to PARA. If [:SOURce#] or # is ignored, then the default is CH1. "1, 2"can only be selected in normal mode; "5" can only be selected in serial mode; "6"can only be selected in parallel mode.

For Example: :SOURce1:VOLTage:PROTection 30.00

[:SOURce#]:VOLTage:PROTection[:LEVel]?

Format: [:SOURce#]:VOLTage:PROTection[:LEVel]?

Function: Query the overvoltage protective value of a certain channel.

Description: The optional value for # are"1|2|3|5|6". 1 is corresponding to CH1; 2 is corresponding to CH2; 3 is corresponding to CH3; 5 is corresponding to SER; 6 is corresponding to PARA. If [:SOURce#] or # is ignored, then the default is CH1.

For Example: :SOURce1:VOLTage:PROTection?

Return: 30.00

[:SOURce#]:VOLTage:PROTection:STATe

Format: [:SOURce#]:VOLTage:PROTection:STATe {0|1|OFF|ON}

Function: Set the overvoltage protective switch of a certain channel and set the channel as the current channel.

Description: The optional value for # are"1|2|3|5|6". 1 is corresponding to CH1; 2 is corresponding to CH2; 3 is corresponding to CH3; 5 is corresponding to SER; 6 is corresponding to PARA. If [:SOURce#] or # is ignored, then the default is CH1. "1, 2"can only be selected in normal mode; "5" can only be selected in serial mode; "6"can only be selected in parallel mode.

For Example: :SOURce1:VOLTage:PROTection:STATe 1

[:SOURce#]:VOLTage:PROTection:STATe?

Format: [:SOURce#]:VOLTage:PROTection:STATe?

Function: Query the overvoltage protective switch of a certain channel.

Description: The optional value for # are"1|2|3|5|6". 1 is corresponding to CH1; 2 is corresponding to CH2; 3 is corresponding to CH3; 5 is corresponding to SER; 6 is corresponding to PARA. If [:SOURce#] or # is ignored, then the default is CH1. It returns"ON | OFF".

For Example: :SOURce1:VOLTage:PROTection:STATe?

Return: ON

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

Format: [:SOURce#]:CURRent[:LEVel][:IMMediate][:AMPLitude] {<curr>|MINimum|MAXimum}

Function: Set the current value of a certain channel and set the channel as the current channel.

Description: The optional value for # are"1|2|3|5|6". 1 is corresponding to CH1; 2 is corresponding to CH2; 3 is corresponding to CH3; 5 is corresponding to SER; 6 is corresponding to PARA. If [:SOURce#] or # is ignored, then the default is CH1. "1, 2"can only be selected in normal mode; "5" can only be selected in serial mode; "6"can only be selected in parallel mode.

For Example: :SOURce1:CURRent 5.000

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

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

Function: Query the current value of a certain channel.

Description: The optional value for # are"1|2|3|5|6". 1 is corresponding to CH1; 2 is corresponding to CH2; 3 is corresponding to CH3; 5 is corresponding to SER; 6 is corresponding to PARA. If [:SOURce#] or # is ignored, then the default is CH1.

For Example: :SOURce1:CURRent?

Return: 5.000

[:SOURce#]:CURRent:PROTection[:LEVel]

Format: [:SOURce#]:CURRent:PROTection[:LEVel] {<curr>|MINimum|MAXimum}

Function: Set the overcurrent protective value of a certain channel and set the channel as the current channel.

Description: The optional value for # are"1|2|3|5|6". 1 is corresponding to CH1; 2 is corresponding to CH2; 3 is corresponding to CH3; 5 is corresponding to SER; 6 is corresponding to PARA. If [:SOURce#] or # is ignored, then the default is CH1. "1, 2"can only be selected in normal mode; "5" can only be selected in serial mode; "6"can only be selected in parallel mode.

For Example: :SOURce1:CURRent:PROTection 5.000

[:SOURce#]:CURRent:PROTection[:LEVel]?

Format: [:SOURce#]:CURRent:PROTection[:LEVel]?

Function: Query the overcurrent protective value of a certain channel.

Description: The optional value for # are"1|2|3|5|6". 1 is corresponding to CH1; 2 is corresponding to CH2; 3 is corresponding to CH3; 5 is corresponding to SER; 6 is corresponding to PARA. If [:SOURce#] or # is ignored, then the default is CH1.

For Example: :SOURce1:CURRent:PROTection?

Return: 5.000

[:SOURce#]:CURRent:PROTection:STATe

Format: **[:SOURce#]:CURRent:PROTection:STATe** {0|1|OFF|ON}

Function: Set the overcurrent protective switch of a certain channel and set the channel as the current channel.

Description: The optional value for # are"1|2|3|5|6". 1 is corresponding to CH1; 2 is corresponding to CH2; 3 is corresponding to CH3; 5 is corresponding to SER; 6 is corresponding to PARA. If [:SOURce#] or # is ignored, then the default is CH1. "1, 2"can only be selected in normal mode; "5" can only be selected in serial mode; "6"can only be selected in parallel mode.

For Example: :SOURce1:CURRent:PROTection:STATe 1

[:SOURce#]:CURRent:PROTection:STATe?

Format: **[:SOURce#]:CURRent:PROTection:STATe?**

Function: Query the overcurrent protective switch of a certain channel.

Description: The optional value for # are"1|2|3|5|6". 1 is corresponding to CH1; 2 is corresponding to CH2; 3 is corresponding to CH3; 5 is corresponding to SER; 6 is corresponding to PARA. If [:SOURce#] or # is ignored, then the default is CH1. It returns"ON | OFF".

For Example: :SOURce1:CURRent:PROTection:STATe?

Return: ON

5. OUTPut

:OUTPut[:STATe]

Format: **:OUTPut[:STATe]** [CH1|CH2|CH3|SER|PARA|ALL,] {0|1|OFF|ON}

Function: Turn on/off output switch of a certain channel and set the channel as the current channel.

Description: If the channel parameter is ignored, then the current channel which is the channel need to be controlled by default. "CH1, CH2"can only be selected in normal mode; "SER" can only be selected in serial mode; "PARA"can only be selected in parallel mode.

For Example: :OUTPut:STATe CH1, ON

:OUTPut[:STATe]?

Format: **:OUTPut[:STATe]?** [CH1|CH2|CH3|SER|PARA]

Function: Query the status of the output swtich of a certain channel.

Description: If the channel parameter is ignored, then the current channel which is the channel need to be queried by default. It returns "ON | OFF".

For Example: :OUTPut:STATe? CH1

Return: ON

:OUTPut:CVCC?

Format: **:OUTPut:CVCC?** [CH1|CH2|CH3|SER|PARA]

Function: Query the status of the constant voltage and constant current of a certain channel.

Description: If the channel parameter is ignored, then the current channel which is the channel need to be queried by default. It returns"CV | CC".

For Example: :OUTPut:CVCC? CH1

Return: CV

:OUTPut:OVP:VALue

Format: **:OUTPut:OVP:VALue** [CH1|CH2|CH3|SER|PARA,] {<voI>|MINimum|MAXimum}

Function: Set the overvoltage protective value of a certain channel and set the channel as the current channel.

Description: If the channel parameter is ignored, then the current channel which is the channel need to be controlled by default. "CH1, CH2"can only be selected in normal mode; "SER" can only be selected in serial mode; "PARA"can only be selected in parallel mode.

For Example: :OUTPut:OVP:VALue CH1, 5

:OUTPut:OVP:VALue?

Format: **:OUTPut:OVP:VALue?** [CH1|CH2|CH3|SER|PARA]

Function: Query the overvoltage protective value of a certain channel.

Description: If the channel parameter is ignored, then the current channel which is the channel need to be queried by default.

For Example: :OUTPut:OVP:VALue? CH1

Return: 5.00

:OUTPut:OVP[:STATe]

Format: **:OUTPut:OVP[:STATe]** [CH1|CH2|CH3|SER|PARA,] {0|1|OFF|ON}

Function: Set the overvoltage protective switch of a certain channel and set the channel as the current channel.

Description: If the channel parameter is ignored, then the current channel which is the channel need to be controlled by default. "CH1, CH2"can only be selected in normal mode; "SER" can only be selected in serial mode; "PARA"can only be selected in parallel mode.

For Example: :OUTPut:OVP:STATe CH1, ON

:OUTPut:OVP[:STATe]?

Format: **:OUTPut:OVP[:STATe]?** [CH1|CH2|CH3|SER|PARA]

Function: Query the overvoltage protective switch of a certain channel.

Description: If the channel parameter is ignored, then the current channel which is the channel need to be queried by default. It return"ON | OFF".

For Example: :OUTPut:OVP:STATe? CH1

Return: ON

:OUTPut:OCP:VALue

Format: **:OUTPut:OCP:VALue** [CH1|CH2|CH3|SER|PARA,] {<curr>|MINimum|MAXimum}

Function: Set the overcurrent value of a certain channel and set the channel as the current channel.

Description: If the channel parameter is ignored, then the current channel which is the channel need to be controlled by default. "CH1, CH2"can only be selected in normal mode; "SER" can only be selected in serial mode; "PARA"can only be selected in parallel mode.

For Example: :OUTPut:OCP:VALue CH1, 5.1

:OUTPut:OCP:VALue?

Format: **:OUTPut:OCP:VALue?** [CH1|CH2|CH3|SER|PARA]

Function: Query the overcurrent value of a certain channel.

Description: If the channel parameter is ignored, then the current channel which is the channel need to be queried by default.

For Example: :OUTPut:OCP:VALue? CH1

Return: 5.100

:OUTPut:OCP[:STATe]

Format: **:OUTPut:OCP[:STATe]** [CH1|CH2|CH3|SER|PARA,] {0|1|OFF|ON}

Function: Set the overcurrent protective switch of a certain channel and set the channel as the current channel.

Description: If the channel parameter is ignored, then the current channel which is the channel need to be controlled by default. "CH1, CH2"can only be selected in normal mode; "SER" can only be selected in serial mode; "PARA"can only be selected in parallel mode.

For Example: :OUTPut:OCP:STATe CH1, ON

:OUTPut:OCP[:STATe]?

Format: **:OUTPut:OCP[:STATe]?** [CH1|CH2|CH3|SER|PARA]

Function: Query the overcurrent protective switch of a certain channel.

Description: If the channel parameter is ignored, then the current channel which is the channel need to be queried by default. It returns"ON | OFF".

For Example: :OUTPut:OCP:STATe? CH1

Return: ON

6. MEASure

:MEASure:ALL[:DC]?

Format: **:MEASure:ALL[:DC]?** [CH1|CH2|CH3|SER|PARA]

Function: Query the actual voltage, current and power value at the output terminal of the specified channel.

Description: If the channel parameter is ignored, then the current channel which is the channel need to be queried by default.

For Example: :MEASure:ALL? CH1

Return: 05.10,0.089,00.45

:MEASure[:VOLTage][:DC]?

Format: **:MEASure[:VOLTage][:DC]?** [CH1|CH2|CH3|SER|PARA]

Function: Query the actual voltage value at the output terminal of the specified channel.

Description: If the channel parameter is ignored, then the current channel which is the channel need to be queried by default.

For Example: :MEASure:VOLTage? CH1

Return: 05.10

:MEASure:CURRent[:DC]?

Format: **:MEASure:CURRent[:DC]?** [CH1|CH2|CH3|SER|PARA]

Function: Query the actual current value at the output terminal of the specified channel.

Description: If the channel parameter is ignored, then the current channel which is the channel need to be queried by default.

For Example: :MEASure:CURRent? CH1

Return: 0.089

:MEASure:POWEr[:DC]?

Format: **:MEASure:POWEr[:DC]?** [CH1|CH2|CH3|SER|PARA]

Function: Query the actual power value at the output terminal of the specified channel.

Description: If the channel parameter is ignored, then the current channel which is the channel need to be queried by default.

For Example: :MEASure:POWEr? CH1

Return: 00.45

7. LISTout

:LISTout[:STATe]

Format: **:LISTout[:STATe]** {0|1|OFF|ON}

Function: Start or stop the list output mode of the current channel.

Description: It can set and query the current channel by the command in"INSTrument"chapter.

For Example: :LISTout:STATe ON

:LISTout[:STATe]?

Format： :LISTout[:STATe]?

Function： Query the list output status mode of the current channel.

Description： It returns"Status (ON|PAUSED|ERROR|OFF), remaining time, the current point, termination point, remaining cycle count, stop status (OFF|LAST)".

For Example： :LISTout[:STATe]?

Return：ON,1.0,7,2047,0,OFF

:LISTout:BASE

Format： :LISTout:BASE <StrN>, <Grpn>, <CycN>, {OFF|LAST}

Function： Set the basic parameter for list output mode of the current channel.

Description： StrN：Output initial group number，range 0-2047，StrN + Grpn can not greater than 2048.

Grpn：Output group number，range 1-2048，StrN + Grpn can not greater than 2048.

CycN: Cycle count，range 1-99999.

OFF|LAST：Stop status, close output or keep the last output status.

Notice：When list output is running, the parameter can not be set.

For Example： :LISTout:BASE 1,100,1,OFF

:LISTout:BASE?

Format： :LISTout:BASE?

Function： Query the basic parameter of list output mode of the current channel.

Description： It returns"StrN, Grpn, CycN, {OFF|LAST}"

StrN：Output initial group number，range 0-2047.

Grpn：Output group number，range 1-2048.

CycN: Cycle count，range 1-99999.

OFF|LAST：Stop status, close output or keep the last output status.

For Example： :LISTout:BASE?

Return：1,100,1,OFF

:LISTout:PARAmeter

Format： :LISTout:PARAmeter <index>, <volt>, <curr>, <time>

Function： Set the group parameter for list output mode of the current channel.

Description： index：The group serial number for the group need to be set，range 0-2047.

volt：Output voltage of the group，unit is volt.

curr: Output current of the group，unit is ampere.

time：Output time of the group，unit is second.

Notice：When list output is running, the parameter can not be set.

For Example： :LISTout:PARAmeter 0, 10.00, 3.00, 1.5

:LISTout:PARAmeter?

Format：:LISTout:PARAmeter? <index>[,<count>]

Function： Query the group parmeter of list output of the current channel.

Description： index：Query the setting parameter start from the serial number of the group ，range 0-2047.

count：Number of group that need to be queried，range 1-10. When this parameter is ignored, query 1 group by default.

Return Format：It returns data by data block，data field is serial number, voltage, current, time of each group. Such as"#2190,10.000,3.000,1.5;"，#219 represents data field have 19 data—"0,10.000,3.000,1.5;", data contents shows group number 0, voltage 10.000, current 3.000，time 1.5s.

For Example： :LISTout:PARAmeter? 0

Return：#2190,10.000,3.000,1.5;

:LISTout:TEMPlet:SELect

Format：:LISTout:TEMPlet:SELect {SINE|PULSE|RAMP|UP|DN|UPDN|RISE|FALL}

Function： Select the template type for list output of the current channel.

Description： SINE：Template of sine wave;

PULSE：Template of pulse mode;

RAMP: Template of rampe mode;

UP：Template of stair-rising;

DN：Template of stair-down;

UPDN：Template of up/down;

RISE: Template of exponential rise;

FALL：Template of exponential fall.

For Example： :LISTout:TEMPlet:SELect SINE

:LISTout:TEMPlet:SELect?

Format：:LISTout:TEMPlet:SELect?

Function： Query the template type of list output of the current channel.

Description： It returns {SINE|PULSE|RAMP|UP|DN|UPDN|RISE|FALL}.

For Example： :LISTout:PARAmeter?

Return：SINE

:LISTout:TEMPlet:OBJect

Format：:LISTout:TEMPlet:OBJect {V|C}

Function： Set the build object of the template for the list output mode of the current channel.

Description： V：Voltage; C：Current;

For Example： :LISTout:TEMPlet:OBJect V

:LISTout:TEMPlet:OBJect?

Format：:LISTout:TEMPlet:OBJect? {V|C}

Function: Query the build object of the template for the list output mode of the current channel.

Description: It returns {V|C}, V: Voltage; C: Current.

For Example: :LISTout:TEMPlet:OBJect?

Return: V

:LISTout:TEMPlet:STARt

Format: :LISTout:TEMPlet:STARt <index>

Function: Set the initial constructed group number of the template for the list output mode of the current channel. And this group is the constructed point of the first template.

Description: index, the initial constructed group, the range is from 0 to (2048 minus the group number that need to construct).

For Example: :LISTout:TEMPlet:STARt 0

:LISTout:TEMPlet:STARt?

Format: :LISTout:TEMPlet:STARt?

Function: Query the initial constructed group number of the template for the list output mode of the current channel. And this group is the constructed point of the first template.

Description: It returns <index>, the initial constructed group, the range is from 0 to (2048 minus the group number that need to construct).

For Example: :LISTout:TEMPlet:STARt?

Return: 0

:LISTout:TEMPlet:POINTs

Format: :LISTout:TEMPlet:POINTs <point>

Function: Set the constructed group number of list output mode for the current channel.

Description: point, the constructed group number, start to constructing from initial group, group number specified by common constructed point, it not chang the setting value of other group. Pulse template needs to constructe 2 groups at least. Other template needs to constructe 10 groups at least.

For Example: :LISTout:TEMPlet:POINTs 50

:LISTout:TEMPlet:POINTs?

Format: :LISTout:TEMPlet:POINTs?

Function: Query the constructed group number of list output mode for the current channel.

Description: It returns < point >, the constructed group.

For Example: :LISTout:TEMPlet:POINTs?

Return: 50

:LISTout:TEMPlet:MAXValue

Format: :LISTout:TEMPlet:MAXValue {<value>|MINimum|MAXimum}

Function: Set the maximum value of the template for the list output mode of the current channel.

Description: value: real value, specifying a number

MINimum: 0

MAXimum: the maximum output value of each channel

For Example: :LISTout:TEMPlet:MAXValue 5.55

:LISTout:TEMPlet:MAXValue?

Format: :LISTout:TEMPlet:MAXValue?

Function: Query the maximum value of the template for the list output mode of the current channel.

Description: It returns < value >, real value.

For Example: :LISTout:TEMPlet:MAXValue?

Return: 5.55

:LISTout:TEMPlet:MINValue

Format: :LISTout:TEMPlet:MINValue {<value>|MINimum|MAXimum}

Function: Set the minimum value of the template for the list output mode of the current channel.

Description: value: real value, specifying a number

MINimum: 0

MAXimum: the maximum output value of each channel

For Example: :LISTout:TEMPlet:MINValue 1.11

:LISTout:TEMPlet:MINValue?

Format: :LISTout:TEMPlet:MINValue?

Function: Query the minimum value of the template for the list output mode of the current channel.

Description: It returns < value >, real value.

For Example: :LISTout:TEMPlet:MINValue?

Return: 1.11

:LISTout:TEMPlet:INTErval

Format: :LISTout:TEMPlet:INTErval <value>

Function: Set the interval time of the template for the list output mode of the current channel.

Description: value: floating-number value, the unit is second, duration time of the specified group parameter, range: 0.1——9999.9.

For Example: :LISTout:TEMPlet:INTErval 1.0

:LISTout:TEMPlet:INTErval?

Format: :LISTout:TEMPlet:INTErval?

Function: Query the interval time of the template for the list output mode of the current channel.

Description: It returns < value >, floating-number value, the unit is second.

For Example: :LISTout:TEMPlet:INTErval?

Return: 1.0

:LISTout:TEMPlet:INVErt

Format： **:LISTout:TEMPlet:INVErt** {0|1|OFF|ON}

Function： Set the phase invert switch of the template for the list output mode of the current channel.

Description： Only the template of sine wave, pulse and rampe have phase invert function. The command can only be used in these template.

For Example： :LISTout:TEMPlet:INVErt ON

:LISTout:TEMPlet:INVErt?

Format： **:LISTout:TEMPlet:INVErt?**

Function： Query the phase invert switch of the template for the list output mode of the current channel.

Description： It returns {OFF|ON }.

For Example： :LISTout:TEMPlet:INVErt?

Return：ON

:LISTout:TEMPlet:WIDTh

Format： **:LISTout:TEMPlet:WIDTh** <value>

Function： Set the pulse width of the pulse width template for the list output mode of the current channel.

Description： value：floating-number value，the unit is second. To specifies pulse width, range：0.1—（period-0.1）. The command can only be used in pulse width template.

For Example： :LISTout:TEMPlet:WIDTh 5.0

:LISTout:TEMPlet:WIDTh?

Format： **:LISTout:TEMPlet:WIDTh?**

Function： Query the pulse width of the pulse width template for the list output mode of the current channel.

Description： It returns <value>，floating-number value，the unit is second.

For Example： :LISTout:TEMPlet:WIDTh?

Return：5.0

:LISTout:TEMPlet:PERIod

Format： **:LISTout:TEMPlet:PERIod** <value>

Function： Set the period of the pulse width template for the list output mode of the current channel.

Description： value：floating-number value，the unit is second. To specifies the period of pulse, range：（pulse width + 0.1）——9999.9. The command can only be used in pulse width template.

For Example： :LISTout:TEMPlet:PERIod 10.

:LISTout:TEMPlet:PERIod?

Format： **:LISTout:TEMPlet:PERIod?**

Function： Query the period of the pulse width template for the list output mode of the current channel.

Description： It returns <value>，floating-number value，the unit is second.

For Example: :LISTout:TEMPlet:PERIod?

Return: 10.0

:LISTout:TEMPlet:SYMMetry

Format: **:LISTout:TEMPlet:SYMMetry** <value>

Function: Set the symmetry of the rampe template for the list output mode of the current channel.

Description: value: integer value, to specifies the symmetry of the rampe, range: 0——100. The command can only be used in rampe template.

For Example: :LISTout:TEMPlet:SYMMetry 50

:LISTout:TEMPlet:SYMMetry?

Format: **:LISTout:TEMPlet:SYMMetry?**

Function: Query the symmetry of the rampe template for the list output mode of the current channel.

Description: It returns <value>, integer value.

For Example: :LISTout:TEMPlet:SYMMetry?

Return: 50

:LISTout:TEMPlet:EXPRate

Format: **:LISTout:TEMPlet:EXPRate** <value>

Function: Set the exponent of the exponential template for the list output mode of the current channel.

Description: value: integer value, range: 0——10. When using the exponential rise template, use this command to set rise exponent; When using the exponential fall template, use this command to set fall exponent. The command can not be used in other template.

For Example: :LISTout:TEMPlet:EXPRate 5

:LISTout:TEMPlet:EXPRate?

Format: **:LISTout:TEMPlet:EXPRate?**

Function: Query the symmetry of the rampe template for the list output mode of the current channel.

Description: It returns <value>, integer value.

For Example: :LISTout:TEMPlet:EXPRate?

Return: 5

:LISTout:TEMPlet:CONSTruct

Format: **:LISTout:TEMPlet:CONSTruct**

Function: Start to constructing the list group parameter.

Description: After the instrument receives the command, start to constructing the group parameter for list output by the template parameter that has been set.

For Example: :LISTout:TEMPlet:CONSTruct

8. DELAY

:DELAY[:STATe]

Format: :DELAY[:STATe] {0|1|OFF|ON}

Function: Start or stop the delayer of the current channel.

Description: It can set and query the current channel by the command in"INSTrument"chapter.

For Example: :DELAY:STATe ON

:DELAY[:STATe]?

Format: :DELAY[:STATe]?

Function: Query the delayer status of the current channel.

Description: It returns"Running status(ON|OFF), remaining time, the current point, termination point, remaining cycle count, stop status (OFF|LAST|ON)".

For Example: :DELAY[:STATe]?

Return: ON,1,12,2047,0,OFF

:DELAY:STARt

Format: :DELAY:STARt <StrN>

Function: Set the initial group number of delayer output for the current channel.

Description: StrN: output the initial group number, integer value, range: 0-2047, (initial group number + output group number) cannot greater than 2048.

Notice: When the delayer is running, the parameter can not be set.

For Example: :DELAY:STARt 0

:DELAY:STARt?

Format: :DELAY:STARt?

Function: Query the initial group number of delayer output for the current channel.

Description: It returns"StrN", integer value.

For Example: :DELAY:STARt?

Return: 0

:DELAY:GROUPs

Format: :DELAY:GROUPs <Grpn>

Function: Set the output group number of delayer for the current channel.

Description: Grpn: output group number, integer value, range: 1-2048, (initial group number + output group number) cannot greater than 2048.

Notice: When the delayer is running, the parameter can not be set.

For Example: :DELAY:GROUPs 100

:DELAY:GROUPs?

Format: :DELAY:GROUPs?

Function: Query the output group number of delayer for the current channel.

Description: It returns"Grpn", integer value.

For Example: :DELAY:GROUPs?

Return: 100

:DELAY:CYCLEs

Format: :DELAY:CYCLEs <CycN>

Function: Query the cycle count of delayer for the current channel.

Description: It returns CycN: cycle count, integer value, range: 1-99999.

Notice: When the delayer is running, the parameter can not be set.

For Example: :DELAY:CYCLEs 1

:DELAY:CYCLEs?

Format: :DELAY:CYCLEs?

Function: Query the stop status of delayer for the current channel.

Description: It returns"CycN: cycle count, integer value.

For Example: :DELAY:CYCLEs?

Return: 1

:DELAY:ENDState

Format: :DELAY:ENDState {ON|OFF|LAST}

Function: Set the stop status of delayer for the current channel.

Description: ON: When the delayer is stop, the output is enabled.

OFF: When the delayer is stop, the output is disabled.

LAST: When the delayer is stop, the last output status will be keeped.

Notice: When the delayer is running, the parameter can not be set.

For Example: :DELAY:ENDState OFF

:DELAY:ENDState?

Format: :DELAY:ENDState?

Function: Query the stop status of delayer for the current channel.

Description: It returns"{ ON|OFF|LAST }".

ON: When the delayer is stop, the output is enabled.

OFF: When the delayer is stop, the output is disabled.

LAST: When the delayer is stop, the last output status will be keeped.

For Example: :DELAY:ENDState?

Return: OFF

:DELAY:STOP

Format: :DELAY:STOP {NONE|<V|>V|<C|>C|<P|>P}[,<value>]

Function: Set the stop condition of delayer for the current channel.

Description: When the delayer is running and meet the stop condition, the delayer will stop automatically. When <value> parameter is ignored, it only set the judgement condition, not change the original judgement value.

Notice: When the delayer is running, the parameter can not be set.

For Example: :DELAY:STOP >V, 10.00

:DELAY:STOP?

Format: :DELAY:STOP?

Function: Query the stop condition of delayer for the current channel.

Description: It returns"(NONE|<V|>V|<C|>C|<P|>P[,<value>])", if the first return parameter is "NONE", it will not have the second parameter.

For Example: :DELAY:STOP?

Return: >V,10.000

:DELAY:PARAmeter

Format: :DELAY:PARAmeter <index>, {ON|OFF}, <time>

Function: Set the group parameter of delayer for the current channel.

Description: index: group serial number of the group that need to be set, range: 0-2047.

ON|OFF: Output status of the group.

time: Running time of the group, the unit is second.

Notice: When the delayer is running, the parameter can not be set.

For Example: :DELAY:PARAmeter 0, ON, 10

:DELAY:PARAmeter?

Format: :DELAY:PARAmeter? <index>[,<count>]

Function: Query the group parameter of delayer for the current channel.

Description: index: Query the setting parameter start from the serial number of the group, range: 0-2047.

count: Number of group that need to be queried, range: 1-10. When the parameter is ignored, query 1 group by default.

Return Format: It returns data by data block, data field is serial number, output switch status and time of each group. Such as "#190,ON,1.0;", #19 represents data field have 19 data ——"0,ON,1.0;", data contents shows group number 0, the output is enabled, time is 1.0s.

For Example: :DELAY:PARAmeter? 0

Return: #190,ON,1.0;

:DELAY:GENerate:STAT

Format: :DELAY:GENerate:STAT < index>, <point>, {01P|10P}

Function: The group parameters of the delayer are automatically generated according to the state generation mode.

Description: After the instrument receives the command, the group parameters of the delayer are automatically generated according to the state generation mode. The parameters of unspecified group are not affected.

index: The group number of the first point, from which generation begins;

point: Point number need to be generated;

01P|10P: It generated by specifies 01 code or 10 code;

Notice: When the delayer is running, the parameter can not be set.

For Example: :DELAY:GENerate:STAT 0, 10, 01P

:DELAY:GENerate:FIX

Format: :DELAY:GENerate:FIX< index>, <point>, <time_on>, <time_off>

Function: The group parameters of the delayer are automatically generated according to the fixed time generation mode.

Description: After the instrument receives the command, the group parameters of the delayer are automatically generated according to the fixed time generation mode. The parameters of unspecified group are not affected.

index: The group number of the first point, from which generation begins;

point: Point number need to be generated;

time_on: Running time for when output status is opend group;

time_off: Running time for when output status is closed group;

Notice: When the delayer is running, the parameter can not be set.

For Example: :DELAY:GENerate:FIX 0, 10, 5, 10

:DELAY:GENerate:INC

Format: :DELAY:GENerate:INC < index>, <point>, <time_base>, <time_step>

Function: The group parameters of the delayer are automatically generated according to single rise generation mode.

Description: After the instrument receives the command, the group parameters of the delayer are automatically generated according to single rise generation mode. The parameters of unspecified group are not affected.

index: The group number of the first point, from which generation begins;

point: Point number need to be generated;

time_base: time base value;

time_step: step value;

Notice: When the delayer is running, the parameter can not be set.

For Example: :DELAY:GENerate:INC 0, 10, 10, 2

:DELAY:GENerate:DEC

Format: :DELAY:GENerate:DEC < index>, <point>, <time_base>, <time_step>

Function: The group parameters of the delayer are automatically generated according to single fall generation

mode.

Description: After the instrument receives the command, the group parameters of the delayer are automatically generated according to single fall generation mode. The parameters of unspecified group are not affected.

index: The group number of the first point, from which generation begins;

point: Point number need to be generated;

time_base: time base value;

time_step: step value;

Notice: When the delayer is running, the parameter can not be set.

For Example: :DELAY:GENerate:DEC 0, 10, 100, 1

:DELAY:GENerate?

Format: :DELAY:GENerate?

Function: Query the automatic generated parameter of delayer for the current channel.

Description: It returns: "STAT, <index>,<point> , {01P|10P}"or

"FIX, <index>,<point> , <time_on>, <time_off>"or

"{INC|DEC}, <index>,<point> , <time_base>, <time_step>"

For Example: :DELAY:GENerate?

Return: DEC,0,10,100,1

9. MONItor

:MONItor[:STATe]

Format: :MONItor[:STATe] {0|1|OFF|ON}

Function: Start or stop the monitor of the current channel.

Description: It can set and query the current channel by the command in"INSTrument"chapter.

For Example: :MONItor:STATe ON

:MONItor[:STATe]?

Format: :MONItor[:STATe]?

Function: Query the monitor status of the current channel.

Description: It returns"{ON|OFF}".

For Example: :MONItor:STATe?

Return: ON

:MONItor:VOLTage

Format: :MONItor:VOLTage {<V|>V|NONE} [,<volt>|MINimum|MAXimum]

Function: Set the voltage condition in the monitor terms for the current channel.

Description: When the second parameter is ignored, it only set judgement condition, not change the original judgement value.

<V: When voltage judgement is enabled, it returns "True" if voltage is less than the judgement value;

>V: When voltage judgement is enabled, it returns "True" if voltage is greater than the judgement value;

NONE: The voltage judgement can be forbidden, but the voltage judgement, current judgement, power judgement cannot be forbidden at the same time;

<volt>: Real value, voltage judgement value.

For Example: :MONItor:VOLTage >V, 15.58

:MONItor:VOLTage?

Format: :MONItor:VOLTage?

Function: Query the voltage condition in the monitor terms of the current channel.

Description: It returns "{<V|>V|NONE}, <volt>"; <volt>: Real value, voltage judgement value.

For Example: :MONItor:VOLTage?

Return: >V,15.58

:MONItor:CURRent

Format: :MONItor:CURRent {<C|>C|NONE}[,<curr>|MINimum|MAXimum]

Function: Set the current condition in the monitor terms for the current channel.

Description: When the second parameter is ignored, it only set judgement condition, not change the original judgement value.

<C: When current judgement is enabled, it returns "True" if current is less than the judgement value;

>C: When current judgement is enabled, it returns "True" if current is greater than the judgement value;

NONE: The current judgement can be forbidden, but the voltage judgement, current judgement, power judgement cannot be forbidden at the same time;

<curr>: Real value, current judgement value.

For Example: :MONItor:CURRent >C, 3.555

:MONItor:CURRent?

Format: :MONItor:CURRent?

Function: Query the current condition in the monitor terms of the current channel.

Description: It returns"{<C|>C|NONE}, <curr>"; <curr>: Real value, current judgement value.

For Example: :MONItor:CURRent?

Return: >C,3.555

:MONItor:POWER

Format: :MONItor:POWER {<P|>P|NONE}[,<watt>|MINimum|MAXimum]

Function: Set the power condition in the monitor terms for the current channel.

Description: When the second parameter is ignored, it only set judgement condition, not change the original judgement value.

<P: When power judgement is enabled, it returns "True" if power is less than the judgement value;

>P: When power judgement is enabled, it returns "True" if power is greater than the judgement value;

NONE: The power judgement can be forbidden, but the voltage judgement, current judgement, power judgement cannot be forbidden at the same time;

<watt>: Real value, power judgement value.

For Example: :MONItor:POWER >P, 60.00

:MONItor:POWER?

Format: :MONItor:POWER?

Function: Query the power condition in the monitor terms of the current channel.

Description: It returns"{<P|>P|NONE}, <watt>"; <watt>: Real value, power judgement value.

For Example: :MONItor:POWER?

Return: >P,60.00

:MONItor:LOGic

Format: :MONItor:LOGic {1|2}, {AND|OR}

Function: Set the logical symbol in the monitor terms for the current channel.

Description: {1|2}: 1 specifies to set the first logical symbol; 2 specifies to set the second logical symbol;

AND: Set the specified logical symbol to "AND";

OR: Set the specified logical symbol to "OR".

For Example: :MONItor:LOGic 1, AND

:MONItor:LOGic 2, OR

:MONItor:LOGic?

Format: :MONItor:LOGic? {1|2}

Function: the logical symbol in the monitor terms of the current channel.

Description: {1|2}: 1 specifies to set the first logical symbol; 2 specifies to set the second logical symbol;

It returns:"{AND|OR}".

For Example: :MONItor:LOGic? 1

Return: AND

:MONItor:LOGic? 2

Return: OR

:MONItor:STOPway

Format: :MONItor:STOPway {OUTOFF|MSG|BEEPER}, {ON|OFF}

Function: Set the stop way of the monitor for the current channel.

Description: OUTOFF: Set to turn off output or not; ON: Turn off output; OFF: Turn on output;

MSG: Set to pop-out hint message or not ON: Pop-out hint message OFF: Not pop-out hint message

BEEPER: Set to turn on/off beeper sound ON: Turn on beeper OFF: Turn off beeper.

For Example: : :MONItor:STOPway OUTOFF, ON Turn off output when the monitoring condition is met.
:MONItor:STOPway MSG, OFF Hint message will not be pop-out when the monitoring condition is met.
:MONItor:STOPway BEEPER, ON The beeper will sound when the monitoring condition is met.

:MONItor:STOPway?

Format: **:MONItor:STOPway?**

Function: Query the stop way of the monitor for the current channel.

Description: It returns:"OutputOff:{ON|OFF}, Msg:{ON|OFF}, Beep:{ON|OFF}".

For Example: :MONItor:STOPway?
Return: OutputOff:ON,Msg:OFF,Beep:ON

10. TRIGger

:TRIGger:IN[:ENABle]

Format: **:TRIGger:IN[:ENABle]** {D0|D1|D2|D3},{0|1|OFF|ON}

Function: Start or stop the trigger of input mode; If the trigger is in output mode, the trigger will switch to input mode.

Description: D0|D1|D2|D3: Select trigger IO, D0-IO1, D1-IO2, D2-IO3, D3-IO4。

For Example: :TRIGger:IN:ENABle D0, OFF Stop the trigger of IO1, IO1 will be the input mode.
:TRIGger:IN:ENABle D0, ON Start the trigger of IO1, IO1 will be the input mode.

:TRIGger:IN[:ENABle]?

Format: **:TRIGger:IN[:ENABle]?** {D0|D1|D2|D3}

Function: Query the trigger status of input mode.

Description: D0|D1|D2|D3: Select trigger IO, D0-IO1, D1-IO2, D2-IO3, D3-IO4。
When IO is in output mode, it must return "OFF";
When IO is in input mode, it will retrun"OFF"or"ON"by the status of trigger.

For Example: :TRIGger:IN:ENABle? D0
Return: ON

:TRIGger:IN:SOURce

Format: **:TRIGger:IN:SOURce** {D0|D1|D2|D3}, {CH1|CH2|CH3|SER|PARA}[, {CH1|CH2|CH3|SER|PARA}][,{CH1|CH2|CH3|SER|PARA}]

Function: Set the controlled source for input mode trigger. When input IO gernerated trigger signal, the controlled source executes the action according to the setting response.

Description: D0|D1|D2|D3: Select trigger IO, D0-IO1, D1-IO2, D2-IO3, D3-IO4.
CH1: Channel 1; CH2: Channel 2; CH3: Channel 3; SER: Serial channel; PARA: Parallel channel.
In the last three parameter, CH1\CH2 cannot show up with SER\PARA at the same time, SER and PARA cannot appears at the same time.

For Example: :TRIGger:IN:SOURce D0, CH1, CH2, CH3 Set the cotrolled source of IO1 to CH1, CH2 and CH3

:TRIGger:IN:SOURce D0, CH1, CH3 Set the cotrolled source of IO1 to CH1 and CH3

:TRIGger:IN:SOURce D0, CH3, SER Set the cotrolled source of IO1 to CH3 and SER(serial channel)

:TRIGger:IN:SOURce D0, PARA Set the cotrolled source of IO1 to PARA(parallel channel)

:TRIGger:IN:SOURce?

Format: :TRIGger:IN:SOURce? {D0|D1|D2|D3}

Function: Query the controlled source of the input mode trigger.

Description: D0|D1|D2|D3: Select trigger IO, D0-IO1, D1-IO2, D2-IO3, D3-IO4.

It returns all combination of controlled channel. Such as: "CH2,CH3", it returns two controlled channel, channel 2 and channel 3.

CH1: Channel 1; CH2: Channel 2; CH3: Channel 3; SER: Serial channel; PARA: Parallel channel.

For Example: :TRIGger:IN:SOURce? d0

Return: CH2,CH3

:TRIGger:IN:TYPE

Format: :TRIGger:IN:TYPE {D0|D1|D2|D3},{RISE|FALL|HIGH|LOW}

Function: Set the trigger signal type for input mode trigger.

Description: D0|D1|D2|D3: Select trigger, IO, D0-IO1, D1-IO2, D2-IO3, D3-IO4.

RISE: Rising edge signal;

FALL: Falling edge signal;

HIGH: High-level signal;

LOW: Low-level signal.

For Example: :TRIGger:IN:TYPE D0, FALL Set the trigger input signal type of IO1 to be the falling edge.

:TRIGger:IN:TYPE?

Format: :TRIGger:IN:TYPE? {D0|D1|D2|D3}

Function: uery the trigger signal type of input mode trigger.

Description: D0|D1|D2|D3: Select trigger IO, D0-IO1, D1-IO2, D2-IO3, D3-IO4.

It returns"{RISE|FALL|HIGH|LOW}".

RISE: Rising edge signal;

FALL: Falling edge signal;

HIGH: High-level signal;

LOW: Low-level signal.

For Example: :TRIGger:IN:TYPE? D0 Query the trigger signal type of IO1.

Return: FALL Return result: falling edge

:TRIGger:IN:SENSitivity

Format: :TRIGger:IN:SENSitivity {D0|D1|D2|D3},{LOW|MID|HIGH}

Function: Set the sensitivity of trigger signal for the input mode trigger.

Description: D0|D1|D2|D3: Select trigger IO, D0-IO1, D1-IO2, D2-IO3, D3-IO4.

LOW: Low sensitivity;

MID: Middle sensitivity;

HIGH: High sensitivity;

For Example: :TRIGger:IN:SENSitivity D0, MID Set the input trigger signal sensitivity of IO1 to middle sensitivity.

:TRIGger:IN:SENSitivity?

Format: **:TRIGger:IN:SENSitivity?** {D0|D1|D2|D3}

Function: Query the sensitivity of trigger signal of the input mode trigger.

Description: D0|D1|D2|D3: Select trigger IO, D0-IO1, D1-IO2, D2-IO3, D3-IO4.

It returns"{LOW|MID|HIGH }".

LOW: Low sensitivity;

MID: Middle sensitivity;

HIGH: High sensitivity;

For Example: :TRIGger:IN:SENSitivity? D0 Query the input trigger signal sensitivity of IO1.

Return: MID Return result: middle sensitivity

:TRIGger:IN:RESPonse

Format: **:TRIGger:IN:RESPonse** {D0|D1|D2|D3},{ON|OFF|ALTER}

Function: Set the trigger response for input mode trigger, when input trigger event is generated, the trigger will execute the trigger response.

Description: D0|D1|D2|D3: Select trigger IO, D0-IO1, D1-IO2, D2-IO3, D3-IO4.

ON: When input trigger event is generated, the trigger will turn on the output of the controlled channel;

OFF: When input trigger event is generated, the trigger will turn off the output of the controlled channel;

ALTER: When input trigger event is generated, the trigger will invert the output of the controlled channel.

For Example: :TRIGger:IN:RESPonse D0, OFF Set the input trigger response of IO1 to OFF.

:TRIGger:IN:RESPonse?

Format: **:TRIGger:IN:RESPonse?** {D0|D1|D2|D3}

Function: Query the trigger response of input mode trigger.

Description: D0|D1|D2|D3: Select trigger IO, D0-IO1, D1-IO2, D2-IO3, D3-IO4.

It returns"{ON|OFF|ALTER }", it represents when input trigger event is generated, the trigger will execute the trigger response.

ON: When input trigger event is generated, the trigger will turn on the output of the controlled channel;

OFF: When input trigger event is generated, the trigger will turn off the output of the controlled channel;

ALTER: When input trigger event is generated, the trigger will invert the output of the controlled channel;

For Example: :TRIGger:IN:RESPonse? D0 Query the input trigger response of IO1.

Return: OFF Return result: When input trigger event is generated, the trigger will turn off the output of the controlled channel.

:TRIGger:OUT[:ENABle]

Format: :TRIGger:OUT[:ENABle] {D0|D1|D2|D3},{0|1|OFF|ON}

Function: Start or stop output mode trigger; If trigger is in input mode, the trigger will switch to output mode.

Description: D0|D1|D2|D3: Select trigger IO, D0-IO1, D1-IO2, D2-IO3, D3-IO4.

For Example: :TRIGger:OUT:ENABle D0, OFF Stop the trigger of IO1, IO1 to be output mode.

:TRIGger:OUT:ENABle D0, ON Start the trigger of IO1, IO1 to be output mode.

:TRIGger:OUT[:ENABle]?

Format: :TRIGger:OUT[:ENABle]? {D0|D1|D2|D3}

Function: Query the status of output mode trigger.

Description: D0|D1|D2|D3: Select trigger IO, D0-IO1, D1-IO2, D2-IO3, D3-IO4.

When IO is in input mode, it must return "OFF";

When IO is in output mode, it will retrun"OFF"or"ON"by the status of trigger.

For Example: :TRIGger:OUT:ENABle? D0

Return: ON

:TRIGger:OUT:SOURce

Format: :TRIGger:OUT:SOURce {D0|D1|D2|D3},{CH1|CH2|CH3|SER|PARA}

Function: Set the controlled source for output mode trigger. IO will output the response signal by the setting output signal when the cotrolled source is meet the trigger condition.

Description: D0|D1|D2|D3: Select output IO, D0-IO1, D1-IO2, D2-IO3, D3-IO4.

Only one channel can select to be the controlled source.

CH1: Channel 1; CH2: Channel 2; CH3: Channel 3; SER: Serial channel; PARA: Parallel channel.

For Example: :TRIGger:OUT:SOURce D0, CH1 Set CH1 to the controlled channel for IO1.

:TRIGger:OUT:SOURce?

Format: :TRIGger:OUT:SOURce? {D0|D1|D2|D3},

Function: Query the controlled source of output mode trigger.

Description: D0|D1|D2|D3: Select output IO, D0-IO1, D1-IO2, D2-IO3, D3-IO4.

Return: {CH1|CH2|CH3|SER|PARA}

CH1: Channel 1; CH2: Channel 2; CH3: Channel 3; SER: Serial channel; PARA: Parallel channel.

For Example: :TRIGger:OUT:SOURce? D0 Query the controlled source of IO1.

Return: CH1 Return result: CH1 is the controlled source of IO1.

:TRIGger:OUT:CONDition

Format: :TRIGger:OUT:CONDition {D0|D1|D2|D3},{AUTO|OUTOFF|OUTON|>V|<V|=V|>C|<C|=C|>P|<P|=P}[,<value>]

Function: Set the trigger condition of output mode trigger, when the controlled source is meet thr trigger condition, IO will output the response signal by the setting output signal.

Description: D0|D1|D2|D3: Select output IO, D0-IO1, D1-IO2, D2-IO3, D3-IO4.

AUTO: Automatic trigger, the instrument can always generate the trigger condition at this time; it cannot have parameter with<value>;

OUTOFF: It will generate the trigger condition when the controlled source turn off the ouput; It cannot have parameter with<value>;

OUTON: It will generate the trigger condition when the controlled source turn on the ouput; It cannot have parameter with<value>;

>V: It will generate the trigger condition when the voltage of the controller source is greater than the judgement value; The parameter <value> cannot be omitted;

<V: It will generate the trigger condition when the voltage of the controller source is less than the judgement value; The parameter <value> cannot be omitted;

=V: It will generate the trigger condition when the voltage of the controller source is equal to the judgement value; The parameter <value> cannot be omitted;

>C: It will generate the trigger condition when the current of the controller source is greater than the judgement value; The parameter <value> cannot be omitted;

<C: It will generate the trigger condition when the current of the controller source is greater than the judgement value; The parameter <value> cannot be omitted;

=C: It will generate the trigger condition when the current of the controller source is equal to the judgement value; The parameter <value> cannot be omitted;

>P: It will generate the trigger condition when the power of the controller source is greater than the judgement value; The parameter <value> cannot be omitted;

<P: It will generate the trigger condition when the power of the controller source is less than the judgement value; The parameter <value> cannot be omitted;

=P: It will generate the trigger condition when the power of the controller source is equal to the judgement value; The parameter <value> cannot be omitted.

For Example: :TRIGger:OUT:CONDition D0,>V,30.00 Set the trigger condition of IO1 to the voltage is greater than 30.00 volts.

:TRIGger:OUT:CONDition

Format: :TRIGger:OUT:CONDition {D0|D1|D2|D3}

Function: Query the trigger condition of output mode trigger.

Description: D0|D1|D2|D3: Select output IO, D0-IO1, D1-IO2, D2-IO3, D3-IO4.

Return: "{AUTO|OUTOFF|OUTON|>V|<V|=V|>C|<C|=C|>P|<P|=P} [,<value>]".

Return"AUTO|OUTOFF|OUTON", it cannot have parameter with<value>;

Return">V|<V|=V|>C|<C|=C|>P|<P|=P", it have parameter<value>.

For Example: :TRIGger:OUT:CONDition? D0 Query the output trigger condition of IO1.

Return: >V,30.00 Return result: trigger condition is voltage is greater than 30.00 volts.

:TRIGger:OUT:POLArity

Format: :TRIGger:OUT:POLArity {D0|D1|D2|D3},{POSItive|NEGAtive}

Function: Set the output signal polarity for output trigger.

Description: D0|D1|D2|D3: Select trigger IO, D0-IO1, D1-IO2, D2-IO3, D3-IO4.

POSItive: Output the positive signal

NEGAtive: Output the negative signal

For Example: :TRIGger:OUT:POLArity D0, POSItive Set the output signal of IO1 to positive.

:TRIGger:OUT:POLArity?

Format: **:TRIGger:OUT:POLArity?** {D0|D1|D2|D3}

Function: Query the output signal polarity of output trigger

Description: D0|D1|D2|D3: Select trigger IO, D0-IO1, D1-IO2, D2-IO3, D3-IO4.

Return: "{POSITIVE|NEGATIVE}".

POSITIVE: Output the positive signal

NEGAtive: Output the negative signal

For Example: :TRIGger:OUT:POLArity? D0 Query the output signal polarity of IO1.

Return: POSITIVE Return result: The output signal polarity of IO1 is positive.

11. PRESet

:PRESet#[:APPLy]

Format: **:PRESet#[:APPLy]**

Function: Apply the preset parameter of the specified group to the output setting parameter.

Description: The optional values for # are"1|2|3|4|5", cannot be omitted and the corresponding preset group.

For Example: :PRESet1:APPLy

:PRESet#:SET:VOLTage

Format: **:PRESet#:SET:VOLTage** {CH1|CH2|CH3|SER|PARA}, {<volt>|MINimum|MAXimum}

Function: Set the voltage value for the preset group.

Description: The optional values for # are"1|2|3|4|5", cannot be omitted and the corresponding preset group.

For Example: :PRESet1:SET:VOLTage CH1, 5.00

:PRESet#:SET:VOLTage?

Format: **:PRESet#:SET:VOLTage?** {CH1|CH2|CH3|SER|PARA}

Function: Query the voltage value of the preset group.

Description: The optional values for # are"1|2|3|4|5", cannot be omitted and the corresponding preset group. It returns the real value of voltage.

For Example: :PRESet1:SET:VOLTage? CH1

Return: 05.00

:PRESet#:SET:CURRent

Format: **:PRESet#:SET:CURRent** {CH1|CH2|CH3|SER|PARA}, {<curr>|MINimum|MAXimum}

Function: Set the current value for the preset group.

Description: The optional values for # are"1|2|3|4|5", cannot be omitted and the corresponding preset group.

For Example: :PRESet1:SET:CURRent CH1, 1.258

:PRESet#:SET:CURRent?

Format: **:PRESet#:SET:CURRent?** {CH1|CH2|CH3|SER|PARA}

Function: Query the current value for the preset group.

Description: The optional values for # are"1|2|3|4|5", cannot be omitted and the corresponding preset group. It returns the real value of voltage.

For Example: :PRESet1:SET:CURRent? CH1

Return: 1.258

:PRESet#:SET:OVP

Format: **:PRESet#:SET:OVP** {CH1|CH2|CH3|SER|PARA}, {0|1|OFF|ON}[,{<volt>|MINimum|MAXimum}]

Function: Set the overvoltage protective switch and value for the preset group.

Description: The optional values for # are"1|2|3|4|5", cannot be omitted and the corresponding preset group. When parameter {<volt>|MINimum|MAXimum} is ignored, it can only set the overvoltage switch, not change the protective value.

For Example: :PRESet1:SET:OVP CH1, OFF

:PRESet1:SET:OVP CH1, ON

:PRESet1:SET:OVP CH1, OFF, 20.00

:PRESet1:SET:OVP CH1, ON, 15.00

:PRESet#:SET:OVP?

Format: **:PRESet#:SET:OVP?** {CH1|CH2|CH3|SER|PARA}

Function: Query the overvoltage protective switch and value of the preset group.

Description: The optional values for # are"1|2|3|4|5", cannot be omitted and the corresponding preset group. It returns"{OFF|ON}, <value>".

OFF|ON: The switch status of overvoltage protective

<value>: Real value, overvoltage protective value

For Example: :PRESet1:SET:OVP? CH1

Return: ON,15.000

:PRESet#:SET:OCP

Format: **:PRESet#:SET:OCP** {CH1|CH2|CH3|SER|PARA}, {0|1|OFF|ON}[,{<curr>|MINimum|MAXimum}]

Function: Set the overcurrent protective switch and value for the preset group.

Description: The optional values for # are"1|2|3|4|5", cannot be omitted and the corresponding preset group. When parameter {<curr>|MINimum|MAXimum} is ignored, it can only set the overcurrent switch, not change the protective value.

For Example: :PRESet1:SET:OCP CH1, OFF

:PRESet1:SET:OCP CH1, ON

:PRESet1:SET:OCP CH1, OFF, 2.000

:PRESet1:SET:OCP CH1, ON, 1.500

:PRESet#:SET:OCP?

Format: **:PRESet#:SET:OCP?** {CH1|CH2|CH3|SER|PARA}

Function: Query the overcurrent protective switch and value of the preset group.

Description: The optional values for # are"1|2|3|4|5", cannot be omitted and the corresponding preset group. It returns"{OFF|ON}, <value>".

OFF|ON: The switch status of overcurrent protective;

<value>: Real value, overcurrent protective value.

For Example: :PRESet1:SET:OCP? CH1

Return: ON,1.500

12. Memory

:MEMory[:STATe]:STORe

Format: **:MEMory[:STATe]:STORe** {STA|LST|DLY}, {1|2|3|4|5|6|7|8|9|10}

Function: Save the setup data to the specified position in internal memory.

Description: "STA"is the setup data of normal mode.LST" is the setup data of list output. "DLY"is the setup data of the delayer.

For Example: :MEM:STOR STA, 1

:MEMory[:STATe]:LOAD

Format: **:MEMory[:STATe]:LOAD** {STA|LST|DLY}, {1|2|3|4|5|6|7|8|9|10}

Function: Loading the setup data of the specified position from the internal memory.

Description: "STA"is the setup data of normal mode."LST" is the setup data of list output."DLY"is the setup data of the delayer.

For Example: :MEM:LOAD STA, 1

:MEMory[:STATe]:DELete

Format: **:MEMory[:STATe]:DELete** {STA|LST|DLY|REC}, {1|2|3|4|5|6|7|8|9|10}

Function: Delete the saved data of the specified position from the internal memory.

Description: "STA"is the setup data of normal mode."LST" is the setup data of list output."DLY"is the setup data of the delayer."REC"is the record file.

For Example: :MEM:DEL STA, 1

:MEMory[:STATe]:VALid?

Format: **:MEMory[:STATe]:VALid?** {STA|LST|DLY|REC}, {1|2|3|4|5|6|7|8|9|10}. RETURN:(YES|NO)

Function: Checke whether data is saved at the specified position internal memory.

Description: STA"is the setup data of normal mode."LST" is the setup data of list output."DLY"is the setup data of the delayer."REC"is the record file.

For Example: :MEM:VAL? STA, 1

Return: NO

13. MMemory

:MMEMory:DISK?

Format: :MMEMory:DISK? RETURN:(NULL | D:\)

Function: Query the drive of external memory.

Description: When the external memory is available, it returns "D:\". When the external memory is not available, it returns"NULL".

For Example: :MMEMory:DISK?

Return: D:\

:MMEMory:CDIRectory

Format: :MMEMory:CDIRectory "<D:\Dir_Name>"

Function: Set the operation path for the current file.

Description: The path length cannot over 200 bytes.

For Example: :MMEMory:CDIRectory "D:\Dir1"

:MMEMory:CDIRectory?

Format: :MMEMory:CDIRectory? RETURN:(NULL | D:\ Dir_Name)

Function: Query the operation path of the current file.

Description: no

For Example: :MMEMory:CDIRectory?

Return: D:\ Dir_Name

:MMEMory:CATalog?

Format: :MMEMory:CATalog? RETURN:(NULL | dir1, dir2, file1)

Function: Query the available file folder and file in the current operation path.

Description: It only returns the file folder and file with English letters and only recognize the following file: status setup file (.sta), file setup file (.lst), delayer setup file (.dly).

For Example: :MMEMory:CATalog?

Return: dir1,file.lst

:MMEMory:STORe

Format: :MMEMory:STORe "<FileName.sta>" Filename extension: (.sta | .lst | .dly)

Function: Save the setup data to the specified file in the current operation path.

Description: The filename extension decides the type of the setup data that will be saved.
status setup file (.sta), file setup file (.lst), delayer setup file (.dly).
For Example: :MMEMory:STORe "FileName.sta"

:MMEMory:LOAD
Format: :MMEMory:LOAD "<FileName.sta>" Filename extension: (.sta | .lst | .dly)
Function: Loading the setup data from the specified file in the current operation path.
Description: The filename extension decides the type of the setup data that will be loaded.
status setup file (.sta), file setup file (.lst), delayer setup file (.dly).
For Example: :MMEMory: LOAD "FileName.sta"

:MMEMory:DELete
Format: :MMEMory:DELete "<FileName.sta>" Filename extension: (.sta | .lst | .dly)
Function: Delete the specified file in the current operation path.
Description: no
For Example: :MMEMory:DEL "FileName.sta"

14. STATus

:STATus:PRESet
Format: :STATus:PRESet
Function: Restore all the enable register and operation register of the suspicious status register to the default status.
Description: no...
For Example: no...

:STATus:OPERation[:EVENt]?
Format: :STATus:OPERation[:EVENt]?
Function: Query the event register of the operation status.
Description: no...
For Example: no...

:STATus:OPERation:CONDition?
Format: :STATus:OPERation:CONDition?
Function: Query the condition register of the operation status.
Description: no...
For Example: no...

:STATus:OPERation:ENABle
Format: :STATus:OPERation:ENABle

Function: Set the enable register for the operation status.

Description: no...

For Example: no...

:STATus:OPERation:ENABle?

Format: :STATus:OPERation:ENABle?

Function: Query the enable register of the operation status.

Description: no...

For Example: no...

:STATus:QUEStionable[:EVENt]?

Format: :STATus:QUEStionable[:EVENt]?

Function: Query the event register of the suspicious status register.

Description: no...

For Example: no...

:STATus:QUEStionable:CONDition?

Format: :STATus:QUEStionable:CONDition?

Function: Query the condition register of the suspicious status register.

Description: no...

For Example: no...

:STATus:QUEStionable:ENABle

Format: :STATus:QUEStionable:ENABle

Function: Set the enable register of the suspicious status register.

Description: no...

For Example: no...

:STATus:QUEStionable:ENABle?

Format: :STATus:QUEStionable:ENABle?

Function: Query the enable register of the suspicious status register.

Description: no...

For Example: no...

:STATus:QUEStionable:INSTrument[:EVENt]?

Format: :STATus:QUEStionable:INSTrument[:EVENt]?

Function: Query the event register of the channel's suspicious status register.

Description: no...

For Example: no...

:STATus:QUEStionable:INSTrument:CONDition?

Format：:STATus:QUEStionable:INSTrument:CONDition?

Function：Query the condition register of the channel's suspicious status register.

Description：no...

For Example：no...

:STATus:QUEStionable:INSTrument:ENABle

Format：:STATus:QUEStionable:INSTrument:ENABle

Function：Query the enable register of the channel's suspicious status register.

Description：no...

For Example：no...

:STATus:QUEStionable:INSTrument:ISUMmary#[:EVENt]?

Format：:STATus:QUEStionable:INSTrument:ISUMmary#[:EVENt]? (1<= # <=6, can be omissible)

Function：Query the event register of suspicious status register of a channel.

Description：# to select the channel，"1"：CH1；"2"：CH2；"3"：CH3；"5"：SER；"6"：PARA

If the parameter is ignored, the currently selected channel is the specified channel by default.

For Example：:STATus:QUEStionable:INSTrument:ISUMmary1? Query the event of CH1.

:STATus:QUEStionable:INSTrument:ISUMmary? Query the event of the current channel.

:STATus:QUEStionable:INSTrument:ISUMmary#:CONDition?

Format：:STATus:QUEStionable:INSTrument:ISUMmary#:CONDition? (1<= # <=6, can be omissible)

Function：Query the condition register of suspicious status register of a channel.

Description：# to select the channel，"1"：CH1；"2"：CH2；"3"：CH3；"5"：SER；"6"：PARA

If the parameter is ignored, the currently selected channel is the specified channel by default.

For Example：:STATus:QUEStionable:INSTrument:ISUMmary1:COND? Query the event of CH1.

:STATus:QUEStionable:INSTrument:ISUMmary#:ENABle

Format：:STATus:QUEStionable:INSTrument:ISUMmary#:ENABle (1<= # <=6, can be omissible)

Function：Query the enable register of suspicious status register of a channel.

Description：# to select the channel，"1"：CH1；"2"：CH2；"3"：CH3；"5"：SER；"6"：PARA

If the parameter is ignored, the currently selected channel is the specified channel by default.

For Example：no...

15. SYSTem

:SYSTem:VERSion?

Format：:SYSTem:VERSion?

Function：Query the version number of SCPI that used by the current system.

Description：no...

For Example: no...

:SYSTem:ERRor[:NEXT]?

Format: :SYSTem:ERRor[:NEXT]?

Function: Read a error message.

Description: no...

For Example: no...

:SYSTem:ERRor:COUNt?

Format: :SYSTem:ERRor:COUNt?

Function: Query the number of error messages that have not yet been read out.

Description: no...

For Example: no...

:SYSTem:BEEPer[:STATe]

Format: :SYSTem:BEEPer[:STATe] {0|1|OFF|ON}

Function: Turn on/off the beeper.

For Example: :SYSTem:BEEPer:STATe OFF Turn off the beeper.

:SYSTem:BEEPer[:STATe]?

Format: :SYSTem:BEEPer[:STATe]?

Function: Query the switch status of the beeper.

Description: It returns"{ON|OFF }".

For Example: :SYSTem:BEEPer:STATe? Query the switch status of the beeper.

Return: OFF Return result: The beeper is disabled.

:SYSTem:BRIGhtness

Format: :SYSTem:BRIGhtness <value>

Function: Set the backlight brightness of LCD.

Description: value, integer value, the range is 1 to 100.

For Example: :SYSTem:BRIGhtness 80 Set the backlight brightness of LCD to 80%.

:SYSTem:BRIGhtness?

Format: :SYSTem:BRIGhtness?

Function: Query the backlight brightness of LCD.

Description: Return< value >, integer value, the range is 1 to 100.

For Example: :SYSTem:BRIGhtness? Query the backlight brightness of LCD.

Return: 80 Return result: The backlight brightness of LCD is 80%.

:SYSTem:COMMunicate:LAN:APPLy

Format： **:SYSTem:COMMunicate:LAN:APPLy**

Function： Apply the network parameters that have been set.

Description： When use the command":SYSTem:COMMunicate:LAN:xxxx"(xxxx represents other network setting command in LAN) to set the network parameter, the setup parameter will not take effect immediately and will not save to nonvolatile memory of the device. It must execute the command ":SYSTem:COMMunicate:LAN:APPLy"to make the parameter validly and save permanently. You can execute multiple commands to modify multiple parameters and then execute a single ":SYSTem:COMMunicate:LAN:APPLy" command together.

For Example： :SYSTem:COMMunicate:LAN:DHCP:STATe ON Turn on DHCPFunction.

:SYSTem:COMMunicate:LAN:APPLy Apply the network parameters that have been set.

:SYSTem:COMMunicate:LAN:DHCP[:STATe]

Format： **:SYSTem:COMMunicate:LAN:DHCP[:STATe]** {0|1|OFF|ON}

Function： Turn on/off DHCPFunction of the network.

Description： When use this command to set the network parameter, the setup parameter will not take effect immediately and will not save to nonvolatile memory of the device. It must execute the command ":SYSTem:COMMunicate:LAN:APPLy"to make the parameter validly and save permanently. You can execute multiple commands to modify multiple parameters and then execute a single ":SYSTem:COMMunicate:LAN:APPLy" command together.

For Example： :SYSTem:COMMunicate:LAN:DHCP:STATe ON Turn on DHCPFunction.

:SYSTem:COMMunicate:LAN:APPLy Apply the network parameters that have been set.

:SYSTem:COMMunicate:LAN:DHCP[:STATe]?

Format： **:SYSTem:COMMunicate:LAN:DHCP[:STATe]?**

Function： Query the switch status of DHCPFunction of the network.

Description： After the network parameter is set and before executing the command":SYSTem:COMMunicate:LAN:APPLy", the result queried by this command is the temporary parameter and it reflects the parameter data used when executing the command ":SYSTem:COMMunicate:LAN:APPLy". Otherwise, the result queried by this command is the currently operating parameter of the device.

It returns"{ON|OFF}".

For Example： :SYSTem:COMMunicate:LAN:DHCP:STATe?

Return：ON

:SYSTem:COMMunicate:LAN:IPADdress

Format： **:SYSTem:COMMunicate:LAN:IPADdress** "x.x.x.x"

Function： Set IP address of the device.

Description: When use this command to set the network parameter, the setup parameter will not take effect immediately and will not save to nonvolatile memory of the device. It must execute the command":SYSTem:COMMunicate:LAN:APPLy"to make the parameter validly and save permanently. You can execute multiple commands to modify multiple parameters and then execute a single ":SYSTem:COMMunicate:LAN:APPLy" command together.

For Example: :SYSTem:COMMunicate:LAN:IPADdress "192.168.10.142" Set IP address to 192.168.10.142.

:SYSTem:COMMunicate:LAN:APPLy Apply the network parameters that have been set.

:SYSTem:COMMunicate:LAN:IPADdress?

Format: :SYSTem:COMMunicate:LAN:IPADdress?

Function: Query IP address of the device.

Description: After the network parameter is set and before executing the command":SYSTem:COMMunicate:LAN:APPLy", the result queried by this command is the temporary parameter and it reflects the parameter data used when executing the command ":SYSTem:COMMunicate:LAN:APPLy". Otherwise, the result queried by this command is the currently operating parameter of the device.

It returns: "192.168.10.142" .

For Example: :SYSTem:COMMunicate:LAN:IPADdress?

Return: "192.168.10.142"

:SYSTem:COMMunicate:LAN:SMASK

Format: :SYSTem:COMMunicate:LAN:SMASK "x.x.x.x"

Function: Set the network subnet mask of the device.

Description: When use this command to set the network parameter, the setup parameter will not take effect immediately and will not save to nonvolatile memory of the device. It must execute the command ":SYSTem:COMMunicate:LAN:APPLy"to make the parameter validly and save permanently. You can execute multiple commands to modify multiple parameters and then execute a single ":SYSTem:COMMunicate:LAN:APPLy" command together.

For Example: :SYSTem:COMMunicate:LAN:SMASK "255.255.255.0" Set the subnet mask to 255.255.255.0.

:SYSTem:COMMunicate:LAN:APPLy Apply the network parameters that have been set.

:SYSTem:COMMunicate:LAN:SMASK?

Format: :SYSTem:COMMunicate:LAN:SMASK?

Function: Query the network subnet mask of the device.

Description: After the network parameter is set and before executing the command":SYSTem:COMMunicate:LAN:APPLy", the result queried by this command is the temporary parameter and it reflects the parameter data used when executing the command

":SYSTem:COMMunicate:LAN:APPLy". Otherwise, the result queried by this command is the currently operating parameter of the device.

It returns: "255.255.255.0" .

For Example: :SYSTem:COMMunicate:LAN:SMASK?

Return: "255.255.255.0"

:SYSTem:COMMunicate:LAN:GATEway

Format: :SYSTem:COMMunicate:LAN:GATEway "x.x.x.x"

Function: Set the network gateway of the device.

Description: When use this command to set the network parameter, the setup parameter will not take effect immediately and will not save to nonvolatile memory of the device. It must execute the command ":SYSTem:COMMunicate:LAN:APPLy"to make the parameter validly and save permanently. You can execute multiple commands to modify multiple parameters and then execute a single ":SYSTem:COMMunicate:LAN:APPLy" command together.

For Example: :SYSTem:COMMunicate:LAN:GATEway "192.168.10.1"

Set the network gateway to 192.168.10.1

:SYSTem:COMMunicate:LAN:APPLy Apply the network parameters that have been set.

:SYSTem:COMMunicate:LAN:GATEway?

Format: :SYSTem:COMMunicate:LAN:GATEway?

Function: Query the network gateway of the device.

Description: After the network parameter is set and before executing the command":SYSTem:COMMunicate:LAN:APPLy", the result queried by this command is the temporary parameter and it reflects the parameter data used when executing the command ":SYSTem:COMMunicate:LAN:APPLy". Otherwise, the result queried by this command is the currently operating parameter of the device.

It returns: "192.168.10.1" .

For Example: :SYSTem:COMMunicate:LAN:GATEway?

Return: "192.168.10.1"

:SYSTem:COMMunicate:RS232:BAUD

Format: :SYSTem:COMMunicate:RS232:BAUD {4800|7200|9600|14400|19200|38400|57600|115200|128000}

Function: Set the baud rate for RS232 interface.

For Example: :SYSTem:COMMunicate:RS232:BAUD 9600 Set the baud rate of RS232 to 9600.

:SYSTem:COMMunicate:RS232:BAUD?

Format: :SYSTem:COMMunicate:RS232:BAUD?

Function: Query the baud rate of RS232 interface.

Description: It returns"{4800|7200|9600|14400|19200|38400|57600|115200|128000} ".

For Example: :SYSTem:COMMunicate:RS232:BAUD? Query the baud rate of RS232 interface.
Return: 9600 Return result: baud rate of RS232 is 9600.

:SYSTem:LOCK

Format: **:SYSTem:LOCK** {0|1|OFF|ON}

Function: Turn on/off the key lock. Long press "LOCK" key to unlock the keyboard.

Description: "ON/OFF"key is valid when the key lock is enabled.

For Example: :SYSTem:LOCK Lock the keyboard.

:SYSTem:LOCK?

Format: **:SYSTem:LOCK?**

Function: Query the status of the key lock.

Description: It returns"{ON|OFF }".

For Example: :SYSTem:LOCK? Query the status of the key lock.
Return: OFF Return result: the keyboard is unlocked.

:SYSTem:KLOCk:STATe

:SYSTem:RWLock[:STATe]

Format: **:SYSTem:KLOCk:STATe** [0|1|OFF|ON]
:SYSTem:RWLock[:STATe] [0|1|OFF|ON]

Function: Turn on/off the remote key lock. The lock can only be unlocked by a remote command or by restarting the device if the remote lock key is enabled.

Description: If the parameter is ignored, the remote key is enabled by default. "ON/OFF"key is valid when the remote key lock is enabled.

For Example: :SYSTem:KLOCk:STAT ON Remote lock
:SYSTem:RWLock Remote lock

:SYSTem:KLOCk:STATe?

:SYSTem:RWLock[:STATe]?

Format: **:SYSTem:KLOCk:STATe?**
:SYSTem:RWLock[:STATe]?

Function: Query the status of the remote key lock.

Description: It returns"{ON|OFF }".

For Example: :SYSTem:KLOCk:STAT? Query the status of the remote key lock.
Return: OFF Return result: remote key lock is disabled.

Programming Example

no...