Instruments.uni-trend.com



UDP5000 Series Programmable Digital Power Supply

SCPI Programming Manual

REV 0

August 2024

Chapter 1 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.

1.1 Command Format

Command consists of a keyword, separator, parameter domain, and end mark. Take the following command as an example:

:VOLTage:LEVel 25

VOLTage, LEVel is keyword, "**:**" and blank is separator, "**25**" is parameter (some commands have multiple parameters and separated by "**,**"), the **line separator** or **carriage return** after the command is the end mark.

For clarity, the following conventions are adopted for subsequent symbols.

- Square Brackets **[ ]**
  The contents (keyword, parameter) in square brackets keywords can be omitted.
- Braces **{ }**
  It encloses a parameter within a command string..
- Angle Braces **< >**
  The parameter enclosed in the angle brackets must be a numerical parameter.
- Vertical Bar **|**
  It separates multiple parameters..
- End Mark
  A line separator <LF> (0x0A) or carriage return <CR> (0x0D).

1.2 Parameter Function

Programming parameters can be numeric, character, Boolean, or other data types Regardless of the type, it is all expressed in ASCII. For more details, see the following table.

Symbol	Meaning	Example
<NR1>	Integer	123, 0123
<NR2>	Fixed floating point number	123., 12.3, 0.123, .123
<NR3>	Floating point number	123, 12.3, 123E+3
<NRF>	It may be <NR1>, <NR2>, or <NR3>	
<Boolean>	Boolean data	0|1|ON|OFF

1.3 Abbreviation

All commands are case-insensitive. The commands can be all input in uppercase letters or in lowercase letters. When using abbreviations, enter all uppercase letters present in the command syntax.

1.4 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 presented by the scientific notation method. The part before e retains three figures behind the decimal point, and the e part retains three figures; the batch return must comply with IEEE 488.2# string data format, **'#'+ the length of character bits** [fixed to one character] + **ASCII of valid data length** + **valid data** + **end mark** ['\n']. Such as, **#3123xxxxxxxxxxxxxxxxxxx\n** represents **123** bytes of valid batch data return format, where '**3**' means that "123" occupies 3 character bits.

Chapter 2 SCPI Command

2.1 Command List

Measurement Command	Function
:MEASure:VOLTage?	Measure the output voltage.
:MEASure:CURRent?	Measure the output current.
:MEASure:POWEr?	Measure the output power.
:MEASure:ALL?	Measure the voltage, current, and power.

Parallel-Connected Measurement Command	Function
:MEASure:PARALLEL:VOLTage?	Measure the voltage in a parallel-connected setup.
:MEASure:PARALLEL:CURRent?	Measure the total current in a parallel-connected setup.
:MEASure:PARALLEL:POWEr?	Measure the total power in a parallel-connected setup.
:MEASure:PARALLEL:ALL?	Measure the output voltage, current, and power in a parallel-connected setup.

Series-Connected Measurement Command	Function
:MEASure:SERIES:VOLTage?	Measure the total voltage in a series-connected setup.
:MEASure:SERIES:CURRent?	Measure the current in a series-connected setup.
:MEASure:SERIES:POWEr?	Measure the total power in a series-connected setup.
:MEASure:SERIES:ALL?	Measure the output voltage, current, and power in a series-connected setup.

Output Setting Command	Function
[:SOURce]:VOLTage[:LEVel][:IMMediate][:AMPLitude]	Set the output voltage value.
[:SOURce]:CURRent[:LEVel][:IMMediate][:AMPLitude]	Set the output current value.
[:SOURce]:VOLTage:PROTection[:LEVel]	Set the OVP (Over-Voltage Protection) value.
[:SOURce]:CURRent:PROTection[:LEVel]	Set the OCP (Over-Current Protection) value.
[:SOURce]:VOLTage:PROTection:STATe	Enable/disable OVP function.
[:SOURce:]CURRent:PROTection:STATe	Enable/disable OCP function.
[:SOURce:]VOLTage:PROTection:TRIPed?	Query the trigger status of the OVP function.
[:SOURce:]VOLTage:PROTection:CLEar	Delete the trigger status of OVP function.
[:SOURce:]CURRent:PROTection:TRIPed?	Query the trigger status of the OCP function.
[:SOURce:]CURRent:PROTection:CLEar	Delete the trigger status of OCP function.
[:SOURce:]VOLTage:STEP	Set the step voltage value.
[:SOURce:]VOLTage:UP	Set the step-up voltage value.
[:SOURce:]VOLTage:DOWN	Set the step-down voltage value.
[:SOURce:]CURRent:STEP	Set the step current value.
[:SOURce:]CURRent:UP	Set the step-up current value.
[:SOURce:]CURRent:DOWN	Set the step-down current value.
[:SOURce:]VOLTage:SLEW:RISing	Set the voltage rise slope.
[:SOURce:]VOLTage:SLEW:FALLing	Set the voltage fall slope.
[:SOURce:]CURRent:SLEW:RISing	Set the current rise slope.
[:SOURce:]CURRent:SLEW:FALLing	Set the current fall slope.
[:SOURce:]RESistance[:LEVel][:IMMediate][:AMPLitude]	Set the internal resistance of the power output.
:OUTPut:MODE	Set the slop mode of the power output.
:OUTPut[:STATe]	Enable/disable power output.
:OUTPut:CVCC?	Query the status of the CV (Constant Voltage) and CC (Constant Current) output.
:OUTPut:OVP:VALue	Set the OVP value.
:OUTPut:OCP:VALue	Set the OCP value.
:OUTPut:OVP[:STATe]	Enable/disable OVP function.
:OUTPut:OCP[:STATe]	Enable/disable OCP function.

:OUTPut:OVP:TRIPed?	Query the trigger status of the OVP function.
:OUTPut:OVP:CLEar	Delete the trigger status of OVP function.
:OUTPut:OCP:TRIPed?	Query the trigger status of the OCP function.
:OUTPut:OCP:CLEar	Delete the trigger status of OCP function.

List Mode Command	Function
:LISTout[:STATe]	Set and query the status of the list mode.
:LISTout:BASE	Set the basic parameter for the list mode.
:LISTout:PARAmeter	Set the group parameter for the list mode.

Delayer Command	Function
:DELAY[:STATe]	Enable/disable delayer.
:DELAY:STARt	Set the start group for the delayer.
:DELAY:GROUPs	Set the group number for the delayer.
:DELAY:CYCLEs	Set the cycle count for the delayer.
:DELAY:ENDState	Set the stop status for the delayer.
:DELAY:STOP	Set the stop condition for the delayer.
:DELAY:PARAmeter	Set the group parameter for the delayer.

System Command	Function
:SYSTem:REMote	Set the power supply to remote control mode.
:SYSTem:LOCal	Set the power supply to panel control mode (local).
:SYSTem:LOCK	Lock the panel.
:SYSTem:UNLOCK	Unlock the panel.
:SYSTem:BEEPer:TEST	Test the beeper: one sound (the beeper must be enabled).
:SYSTem:BEEPer[:STATe]	Enable/disable beeper sound.
:SYSTem:BRIGhtness	Set the screen brightness.
:SYSTem:LANGuage	Set the system language.

Power Output Command	Function
:SYSTem:POWER:POWERDown[:STATe]	Set the power failure detection status.
:SYSTem:POWER:MODE	Set the operation mode for the power supply.
:SYSTem:POWER:ID	Set the power supply ID.
:SYSTem:POWER:POWEROut	Set the power-on hold status for the power supply.
:SYSTem:POWER:OVPDelay	Set the OVP delay for the power supply.
:SYSTem:POWER:OCPDelay	Set the OCP delay for the power supply.
:SYSTem:POWER:ELOAD[:STATe]	Set the discharge load status for the power supply.
Network Setting Command	Function
:SYSTem:COMMunicate:LAN:APPLy	Save and apply configured network parameters.
:SYSTem:COMMunicate:LAN:DHCP[:STATe]	Enable/disable DHCP (Dynamic host configuration protocol).
:SYSTem:COMMunicate:LAN:IPADdress	Set/query the IP address of the LAN port.
:SYSTem:COMMunicate:LAN:SMASK	Set/query the subnet mask of the LAN port.
:SYSTem:COMMunicate:LAN:GATEway	Set/query the gateway address of LAN port.

SCPI Command	Function
:SYSTem:ERRor[:NEXT]?	Query error code and character string in SCPI.
:SYSTem:ERRor:COUNt?	Query error code and character string in SCPI.
:SYSTem:VERSion?	Query the SCPI version number.
*IDN?	Query the instrument information.
*STB?	Query the status byte event register.
*SRE	Set the status byte event enable register in SCPI.
*ESR?	Query the event register in SCPI.
*ESE	Set the event enable register in SCPI.
:STATus:QUEStionable[:EVENt]?	Query QUSE status event register in SCPI.

:STATus:QUEStionable:CONDition?	Query QUSE status event register in SCPI and return the status.
:STATus:QUEStionable:ENABle	Set QUSE status event enable register in SCPI.

2.2 Command Parsing

Command	:MEASure:VOLTage?
Function	Measure the output voltage.
Syntax	:MEASure:VOLTage?
Example	:MEASure:VOLTage? -> <OutVoltage>

Command	:MEASure:CURRent?
Function	Measure the output current.
Syntax	:MEASure:CURRent?
Example	:MEASure:CURRent? -> <OutCurrent>

Command	:MEASure:POWEr?
Function	Measure the output power.
Syntax	:MEASure:POWEr?
Example	:MEASure:POWEr? -> <OutPower>

Command	:MEASure:ALL?
Function	Measure the output voltage, current, and power.
Syntax	:MEASure:ALL?
Example	:MEASure:ALL? -> <OutVoltage>,<OutCurrent>,<OutPower>

Command	:MEASure:PARALLEL:VOLTage?
Function	Measure the voltage in a parallel-connected setup.
Syntax	:MEASure:PARALLEL:VOLTage?
Example	:MEASure:PARALLEL:VOLTage? -> <OutVoltage>

Command	:MEASure:PARALLEL:CURRent?
Function	Measure the total current in a parallel-connected setup.
Syntax	:MEASure:PARALLEL:CURRent?
Example	:MEASure:PARALLEL:CURRent? -> <OutCurrent>

Command	:MEASure:PARALLEL:POWEr?
Function	Measure the total power in a parallel-connected setup.
Syntax	:MEASure:PARALLEL:POWEr?

Example	:MEASure:PARALLEL:POWEr? -> <OutPower>
Command	:MEASure:PARALLEL:ALL?
Function	Measure the output voltage, current, and power in a parallel-connected setup.
Syntax	:MEASure:PARALLEL:ALL?
Example	:MEASure:PARALLEL:ALL? -> <OutVoltage>,<OutCurrent>,<OutPower>

Command	:MEASure:SERIES:VOLTage?
Function	Measure the total voltage in a series-connected setup.
Syntax	:MEASure:SERIES:VOLTage?
Example	:MEASure:SERIES:VOLTage? -> <OutVoltage>

Command	:MEASure:SERIES:CURRent?
Function	Measure the current in a series-connected setup.
Syntax	:MEASure:SERIES:CURRent?
Example	:MEASure:SERIES:CURRent? -> <OutCurrent>

Command	:MEASure:SERIES:POWEr?
Function	Measure the total power in a series-connected setup.
Syntax	:MEASure:SERIES:POWEr?
Example	:MEASure:SERIES:POWEr? -> <OutPower>

Command	:MEASure:SERIES:ALL?
Function	Measure the output voltage, current, and power in a series-connected setup.
Syntax	:MEASure:SERIES:ALL?
Example	:MEASure:SERIES:ALL? -> <OutVoltage>,<OutCurrent>,<OutPower>

Command	[:SOURce]:VOLTage[:LEVel]
Function	Set the output voltage value.
Syntax	[:SOURce]:VOLTage[:LEVel]{<Value>|MINimum|MAXimum}
	[:SOURce]:VOLTage[:LEVel]?
	<Value>= Output voltage setting value, measured in volts (V).
Example	:VOLTage <Value>
	:VOLTage? -> <Value>

Command	[:SOURce]:VOLTage:PROTection[:LEVel]
Function	Set the OVP (Over-Voltage Protection) value.
Syntax	[:SOURce]:VOLTage:PROTection[:LEVel]{<Value>|MINimum|MAXimum}
	[:SOURce]:VOLTage:PROTection[:LEVel]?

	<Value>= OVP value, measured in volts (V).
Example	:VOLTage:PROTection <Value>
	:VOLTage:PROTection? -> <Value>

Command	[:SOURce]:VOLTage:PROTection:STATe
Function	Enable/disable OVP function.
Syntax	[:SOURce]:VOLTage:PROTection:STATe {<Boolean>}
	[:SOURce]:VOLTage:PROTection:STATe?
	<Boolean>=ON|OFF|0|1
Example	:VOLTage:PROTection:STATe <Boolean>
	:VOLTage:PROTection:STATe? -> <Boolean>

Command	[:SOURce:]VOLTage:PROTection:TRIPed?
Function	Query the trigger status of the OVP function.
Syntax	[:SOURce:]VOLTage:PROTection:TRIPed? <State>
	<State>=1: Triggered, 0: Untriggered
Example	:VOLTage:PROTection:TRIPed? -> <State>

Command	[:SOURce:]VOLTage:PROTection:CLEar
Function	Delete the trigger status of OVP function.
Syntax	[:SOURce:]VOLTage:PROTection:CLEar

Command	[:SOURce]:CURRent[:LEVel]
Function	Set the output current value.
Syntax	[:SOURce]:CURRent[:LEVel]{<Value>|MINimum|MAXimum}
	[:SOURce]:CURRent[:LEVel]?
	<Value>= OCP value, measured in amperes (A).
Example	:CURRent <Value>
	:CURRent? -> <Value>

Command	[:SOURce]:CURRent:PROTection[:LEVel]
Function	Set the OVP (Over-Voltage Protection) value.
Syntax	[:SOURce]:CURRent:PROTection[:LEVel]{<Value>|MINimum|MAXimum}
	[:SOURce]:CURRent:PROTection[:LEVel]?
	<Value>= OCP value, measured in amperes (A).
Example	:CURRent:PROTection <Value>
	:CURRent:PROTection? -> <Value>

Command	[:SOURce]:CURRent:PROTection:STATe
Function	Enable/disable OCP function.

Syntax	[:SOURce]:CURRent:PROTection:STATe {<Boolean>}
	[:SOURce]:CURRent:PROTection:STATe?
	<Boolean>=ON|OFF|0|1
Example	:CURRent:PROTection:STATe <Boolean>
	:CURRent:PROTection:STATe? -> <Boolean>

Command	[:SOURce:]CURRent:PROTection:TRIPed?
Function	Query the trigger status of the OCP function.
Syntax	[:SOURce:]CURRent:PROTection:TRIPed? <State>
	<State>=1: Triggered, 0: Untriggered
Example	:CURRent:PROTection:TRIPed? -> <State>

Command	[:SOURce:]CURRent:PROTection:CLEar
Function	Delete the trigger status of OCP function.
Syntax	[:SOURce:]CURRent:PROTection:CLEar

Command	[:SOURce:]VOLTage:STEP
Function	Set the step voltage value.
Syntax	[:SOURce:]VOLTage:STEP {<Value>}
	[:SOURce:]VOLTage:STEP?
	<Value>= Step voltage value, measured in volts (V).
Example	:VOLTage:STEP <Value>
	:VOLTage:STEP? -> <Value>

Command	[:SOURce:]VOLTage:UP
Function	Set the step-up voltage value.
Syntax	[:SOURce:]VOLTage:UP

Command	[:SOURce:]VOLTage:DOWN
Function	Set the step-down voltage value.
Syntax	[:SOURce:]VOLTage:DOWN

Command	[:SOURce:]CURRent:STEP
Function	Set the step current value.
Syntax	[:SOURce:]CURRent:STEP {<Value>}
	[:SOURce:]CURRent:STEP?
	<Value>= Step current value, measured in amperes (A).
Example	:CURRent:STEP <Value>
	:CURRent:STEP? -> <Value>

Command	[:SOURce:]CURRent:UP
Function	Set the step-up current value.
Syntax	[:SOURce:]CURRent:UP

Command	[:SOURce:]CURRent:DOWN
Function	Set the step-down current value.
Syntax	[:SOURce:]CURRent:DOWN

Command	[:SOURce:]VOLTage:SLEW:RISing
Function	Set the voltage rise slope.
Syntax	[:SOURce]:VOLTage:SLEW:RISing {<Value>}
	[:SOURce]:VOLTage:SLEW:RISing?
	<Value>= Voltage rise slope, measured in volts per second (V/s).
Example	:VOLTage:SLEW:RISing <Value>
	:VOLTage:SLEW:RISing? -> <Value>

Command	VOLTage:SLEW:FALLing
Function	Set the voltage fall slope.
Syntax	[:SOURce]:VOLTage:SLEW:FALLing {<Value>}
	[:SOURce]:VOLTage:SLEW:FALLing ?
	<Value>= Voltage rise slope, measured in volts per second (V/s).
Example	:VOLTage:SLEW:FALLing <Value>
	:VOLTage:SLEW:FALLing? -> <Value>

Command	[:SOURce:]CURRent:SLEW:RISing
Function	Set the current rise slope.
Syntax	[:SOURce]:CURRent:SLEW:RISing {<Value>}
	[:SOURce]:CURRent:SLEW:RISing?
	<Value>= Current rise slope, measured in amperes per second (A/s).
Example	:CURRent:SLEW:RISing <Value>
	:CURRent:SLEW:RISing? -> <Value>

Command	[:SOURce:]CURRent:SLEW:FALLing
Function	Set the current fall slope.
Syntax	[:SOURce]:CURRent:SLEW:FALLing {<Value>}
	[:SOURce]:CURRent:SLEW:FALLing?
	<Value>= Current rise slope, measured in amperes per second (A/s).
Example	:CURRent:SLEW:FALLing <Value>
	:CURRent:SLEW:FALLing? -> <Value>

Command	:OUTPut:MODE
Function	Set the slop mode of the power output.
Syntax	:OUTPut:MODE {<Mode>}
	:OUTPut:MODE?
	<Mode>:
	VHS: High-speed priority in CV (Constant Voltage) mode. IHS: High-speed priority in CC (Constant Current) mode.
	VSR: Slope priority in CV (Constant Voltage) mode. ISR: Slope priority in CC (Constant Current) mode.
Example	:OUTPut:MODE <Mode>
	:OUTPut:MODE? -> <Mode>

Command	[:SOURce:]RESistance[:LEVel][:IMMediate][:AMPLitude]
Function	Set the internal resistance of the power output.
Syntax	[:SOURce:]RESistance[:LEVel][:IMMediate][:AMPLitude]{<Value>}
	[:SOURce:]RESistance[:LEVel][:IMMediate][:AMPLitude]?
	<Value> = Power output internal resistance, measured in ohms (Ω), range: [0, 1] Ω.
	Note: The power output internal resistance only functions in normal mode.
Example	:RESistance <Value>
	:RESistance? -> <Value>

Command	:OUTPut[:STATe]
Function	Enable/disable power output.
Syntax	:OUTPut[:STATe]{<Boolean>}
	:OUTPut[:STATe]?
	<Boolean>=ON|OFF|0|1
Example	:OUTPut ON
	:OUTPut? -> <Boolean>

Command	:OUTPut:CVCC?
Function	Query the status of the CV (Constant Voltage) and CC (Constant Current) output.
Syntax	:OUTPut:CVCC? {CC|CV}
	CC= Constant Current CV= Constant Voltage
Example	:OUTPut:CVCC? -> <CC|CV>

Command	:OUTPut:OVP:VALue
Function	Set the OVP value.

Syntax	:OUTPut:OVP:VALue {<Value>|MINimum|MAXimum}
	:OUTPut:OVP:VALue?
	<Value>= OVP value, measured in volts (V).
Example	:OUTPut:OVP:VALue <Value>
	:OUTPut:OVP:VALue? -> <Value>

Command	:OUTPut:OVP:TRIPed?
Function	Query the trigger status of the OVP function.
Syntax	:OUTPut:OVP:TRIPed? -><State>
	<State>=1: Triggered, 0: Untriggered
Example	:OUTPut:OVP:TRIPed? -> <State>

Command	:OUTPut:OVP:CLEar
Function	Delete the trigger status of OVP function.
Syntax	:OUTPut:OVP:CLEar

Command	:OUTPut:OCP:VALue
Function	Set the OCP value.
Syntax	:OUTPut:OCP:VALue {<Value>|MINimum|MAXimum}
	:OUTPut:OCP:VALue?
	<Value>= OCP value, measured in amperes (A).
Example	:OUTPut:OCP:VALue <Value>
	:OUTPut:OCP:VALue? -> <Value>

Command	:OUTPut:OCP[:STATe]
Function	Enable/disable OCP function.
Syntax	:OUTPut:OCP[:STATe]{<Boolean>}
	:OUTPut:OCP[:STATe]?
	<Boolean>=ON|OFF|0|1
Example	:OUTPut:OCP <Boolean>
	:OUTPut:OCP? -> <Boolean>

Command	:OUTPut:OCP:TRIPed?
Function	Query the trigger status of the OCP function.
Syntax	:OUTPut:OCP:TRIPed? -><State>
	<State>=1: Triggered, 0: Untriggered
Example	:OUTPut:OCP:TRIPed? -> <State>

Command	:OUTPut:OCP:CLEar
Function	Delete the trigger status of OCP function.

Syntax	:OUTPut:OCP:CLEar

Command	:LISTout[:STATe]
Function	Set and query the status of the list mode.
Syntax	:LISTout[:STATe]{<Boolean>}
	:LISTout[:STATe]? ->{<State>,<time>,<curGroup>,<endGroup>,<remainCycle>,<endState>}
	<Boolean>=ON|OFF|0|1
	<State>=ON|OFF|COMPLETED|PAUSED
	<time> = Remaining time of the current group, measured in seconds (s).
	<curGroup>= Current group
	<endGroup>= End group
	<remainCycle>= Remain cycle
	<endState>= End states (OFF: Disable output, LAST: Hold output)
Example	:LISTout <Boolean>
	:LISTout? -> ON,0.1,000,009,00000, OFF

Command	:LISTout:BASE
Function	Set the basic parameter for the list mode.
Syntax	:LISTout:BASE {<Start>,<Groups>,<Cycle>,<endState>}
	:LISTout:BASE?->{<Start>,<Groups>,<Cycle>,<endState>}
	<Start>= Start group
	<Groups>= Group number
	<Cycle>= Cycle count, 0 indicates an infinite cycle.
	<endState>= End states (OFF: Disable output, LAST: Hold output)
Example	:LISTout:BASE <Start>,<Groups>,<Cycle>,<endState>
	:LISTout:BASE? -><Start>,<Groups>,<Cycle>,<endState>

Command	:LISTout:PARAmeter
Function	Set the group parameter for the list mode.
Syntax	:LISTout:PARAmeter {<No>,<Volt>,<Curr>,<Time>}
	:LISTout:PARAmeter?->{<No>,<Volt>,<Curr>,<Time>}
	<No>= Group serial number
	<Volt>= Voltage, measured in volts (V).
	<Curr>= Current, measured in amperes (A).
	<Time>= Duration, measured in seconds (s).
Example	:LISTout:PARAmeter <No>,<Volt>,<Curr>,<Time>
	:LISTout:PARAmeter? 0,2 ->#226000,10.000,12.000, 100.0;#226001,20.000,07.539, 2.0;

	#226 indicates that 26 occupies two bytes. The data segment contains 26 values: {000, 10.000, 12.000, 100.0}, which represent 0 V, 10 V, 12 A, and 100 seconds, respectively. For the specific format, refer to the Data Return section in the first chapter.

Command	:DELAY[:STATe]
Function	Set and query the state of the delayer.
Syntax	:DELAY[:STATe]{<Boolean>}
	:DELAY[:STATe]? ->{<State>,<time>,<curGroup>,<endGroup>,<remainCycle>,<endState>}
	<Boolean>=ON|OFF|0|1
	<State>=ON|OFF|FAILED|COMPLETED
	<time>= Remaining time of the current group, measured in seconds (s).
	<curGroup>= Current group
	<endGroup>= Stop group
	<remainCycle>= Remain cycle
	<endState>= End states (OFF: Disable output, LAST: Hold output, ON: Enable output)
Example	:DELAY <Boolean>
	:DELAY? -> ON, 0.4,010,016,99999,OFF

Command	:DELAY:STARt
Function	Set the start group for the delayer.
Syntax	:DELAY:STARt {<Start>}
	:DELAY:STARt?->{<Start>}
	<Start>= The start group of the delayer
Example	:DELAY:STARt 0
	:DELAY:STARt?->0

Command	:DELAY:GROUPs
Function	Set the group number for the delayer.
Syntax	:DELAY:GROUPs {<Groups>}
	:DELAY:GROUPs?->{<Groups>}
	<Groups>= The group number of the delayer
Example	:DELAY:GROUPs 64
	:DELAY:GROUPs?->64

Command	:DELAY:CYCLEs
Function	Set the cycle count for the delayer.
Syntax	:DELAY:CYCLEs {<Cycles>}

	:DELAY:CYCLEs?->{<Cycles>}
	<Cycles>= The cycle count of the delayer, 0 indicates an infinite cycle.
Example	:DELAY:CYCLEs 0
	:DELAY:CYCLEs?->0

Command	:DELAY:ENDState
Function	Set the stop status for the delayer.
Syntax	:DELAY:ENDState {<endState>}
	:DELAY:ENDState?->{<endState>}
	<endState>= End states (OFF: Disable output, LAST: Hold output, ON: Enable output)
Example	:DELAY:ENDState OFF
	:DELAY:ENDState?->OFF

Command	:DELAY:STOP
Function	Set the stop status for the delayer.
Syntax	:DELAY:STOP {NONE|<V|>V|<C|>C|<P|>P [,<Value>]}
	:DELAY:STOP?->{NONE|<V|>V|<C|>C|<P|>P [,<Value>]}
	<Value> = Condition parameter, which can be omitted. The value remains unchanged when omitted.
Example	:DELAY:STOP >V,15.000
	:DELAY:STOP?->V,15.000

Command	:DELAY:PARAmeter
Function	Set the group parameter for the delayer.
Syntax	:DELAY:PARAmeter {<No>,<Boolean>,<Time>}
	:DELAY:PARAmeter?->{<No>,<Boolean>,<Time>}
	<No>= Group serial number
	<Boolean>= Output states (ON: Enable output, OFF: Disable output)
	<Time>= Duration, measured in seconds (s), with a minimum resolution of 1 second.
Example	:DELAY:PARAmeter 0,ON,1.0
	:DELAY:PARAmeter? 0,1 ->#215000,OFF 1.0;
	#215 indicates that 15 occupies two bytes. The data segment contains 15 data values: {000,OFF 1.0;} which 0 represents that the output is enabled, 1.0 represents a duration of 1 seconds. For the specific format, refer to the Data Return section in the first chapter.

Command	:SYSTem:REMote

Function	Set the power supply to remote control mode. The keys will be locked in remote control mode. To return to panel control mode, manually unlock the keys.

Command	:SYSTem:LOCal
Function	Set the power supply to panel control mode.

Command	:SYSTem:LOCK
Function	Lock the control panel.

Command	:SYSTem:UNLOCK
Function	Unlock the control panel.

Command	:SYSTem:BEEPer:TEST
Function	Test the beeper. The beeper will make a sound when the command is sent (the command is only activated when the beeper is enabled).

Command	:SYSTem:BEEPer[:STATe]
Function	Enable/disable beeper sound.
Syntax	:SYSTem:BEEPer[:STATe]{<Boolean>}
	:SYSTem:BEEPer[:STATe]?->{<Boolean>}
	<Boolean>=ON: Enable beeper, OFF: Disable beeper
Example	:SYSTem:BEEPer:STATe ON
	:SYSTem:BEEPer:STATe?->ON

Command	:SYSTem:BRIGhtness
Function	Set the screen brightness.
Syntax	:SYSTem:BRIGhtness {<Value>}
	:SYSTem:BRIGhtness?->{<Value>}
	<Value>= Screen brightness, range: 20 to 100
Example	:SYSTem:BRIGhtness 50
	:SYSTem:BRIGhtness?->50

Command	:SYSTem:LANGuage
Function	Set the system language.
Syntax	:SYSTem:LANGuage {<Language>}
	:SYSTem:LANGuage?->{<Language>}
	<Language>= System language (EN: English, CH: Simplified Chinese)
Example	:SYSTem:LANGuage CH
	:SYSTem:LANGuage?->CH

Command	:SYSTem:POWER:POWERDown[:STATe]
Function	Set the power failure detection status.
Syntax	:SYSTem:POWER:POWERDown[:STATe]{<Boolean>}
	:SYSTem:POWER:POWERDown[:STATe]?->{<Boolean>}
	<Boolean>=ON: Enable the power failure detection, OFF: Disable the power failure detection
Example	:SYSTem:POWER:POWERDown ON
	:SYSTem:POWER:POWERDown?->ON

Command	:SYSTem:POWER:MODE
Function	Set the operation mode for the power supply.
Syntax	:SYSTem:POWER:MODE {<Mode>}
	:SYSTem:POWER:MODE?-> {<Mode>}
	<Mode>= Normal mode, EXT_V: External control
	PARAMaster: Parallel-connected master PARASlave: Parallel-connected slave
	SERMaster: Series-connected master SERSlave: Series-connected slave
Example	:SYSTem:POWER:MODE Normal
	:SYSTem:POWER:MODE?-> Normal

Command	:SYSTem:POWER:ID
Function	Set the power supply ID
Syntax	:SYSTem:POWER:ID {<DeviceId>}
	:SYSTem:POWER:ID?-> {<DeviceId>}
	<DeviceId>: Device ID. In series-connected mode, range: [1,1]; in parallel-connected mode, range: [1,2]
Example	:SYSTem:POWER:ID 1
	:SYSTem:POWER:ID? ->1

Command	:SYSTem:POWER:POWEROut
Function	Set the power-on hold status for the power supply.
Syntax	:SYSTem:POWER:POWEROut {<P-Out>}
	:SYSTem:POWER:POWEROut?-> {<P-Out>}
	{<P-Out>}: KEEP: Retain the output status before the last power-off. OFF: Indicates no output after power-off.
Example	:SYSTem:POWER:POWEROut KEEP
	:SYSTem:POWER:POWEROut?->KEEP

Command	:SYSTem:POWER:OVPDelay
Function	Set the OVP delay for the power supply.
Syntax	:SYSTem:POWER:OVPDelay {<delayTime>}
	:SYSTem:POWER:OVPDelay?-> {<delayTime>}
	{<delayTime>}: OVP delay time, measured in milliseconds (ms).
Example	:SYSTem:POWER:OVPDelay 10
	:SYSTem:POWER:OVPDelay?->10

Command	:SYSTem:POWER:OCPDelay
Function	Set the OCP delay for the power supply.
Syntax	:SYSTem:POWER:OCPDelay {<delayTime>}
	:SYSTem:POWER:OCPDelay?-> {<delayTime>}
	{<delayTime>}: OCP delay time, measured in milliseconds (ms).
Example	:SYSTem:POWER:OCPDelay 10
	:SYSTem:POWER:OCPDelay?->10

Command	:SYSTem:POWER:ELOAD[:STATe]
Function	Set the discharge load status for the power supply.
Syntax	:SYSTem:POWER:ELOAD[:STATe] {<Boolean>}
	:SYSTem:POWER:ELOAD[:STATe]?->{<Boolean>}
	<Boolean>= ON: Enable the discharge load function, OFF: Disable the discharge load function, AUTO: Automatically enable or disable the discharge load
Example	:SYSTem:POWER:ELOAD ON
	:SYSTem:POWER:ELOAD?->ON

Command	:SYSTem:COMMunicate:LAN:APPLy
Function	Save and apply configured network parameters.
	Note: The network settings will not take effect after configuration; they will only be applied when the command is executed.
Syntax	:SYSTem:COMMunicate:LAN:APPLy

Command	:SYSTem:COMMunicate:LAN:DHCP[:STATe]
Function	Enable/disable DHCP.
Syntax	:SYSTem:COMMunicate:LAN:DHCP[:STATe] {<Boolean>}
	:SYSTem:COMMunicate:LAN:DHCP[:STATe]?->{<Boolean>}
	<Boolean>= ON: Enable DHCP, OFF: Disable DHCP
Example	:SYSTem:COMMunicate:LAN:DHCP ON
	:SYSTem:COMMunicate:LAN:APPLy
	:SYSTem:COMMunicate:LAN:DHCP?->ON

Command	:SYSTem:COMMunicate:LAN:IPADdress
Function	Set/query the IP address of the LAN port.
Syntax	:SYSTem:COMMunicate:LAN:IPADdress {<Address>}
	:SYSTem:COMMunicate:LAN:IPADdress?->{<Address>}
	<Address>= IP address in the format "x.x.x.x"
Example	:SYSTem:COMMunicate:LAN:IPADdress "192.168.1.100"
	:SYSTem:COMMunicate:LAN:APPLy
	:SYSTem:COMMunicate:LAN:IPADdress?->192.168.1.100

Command	:SYSTem:COMMunicate:LAN:SMASK
Function	Set/query the subnet mask of the LAN port.
Syntax	:SYSTem:COMMunicate:LAN:SMASK {<Address>}
	:SYSTem:COMMunicate:LAN:SMASK?->{<Address>}
	<Address>= Subnet mask in the format "x.x.x.x"
Example	:SYSTem:COMMunicate:LAN:SMASK "255.255.255.0"
	:SYSTem:COMMunicate:LAN:APPLy
	:SYSTem:COMMunicate:LAN:SMASK?->255.255.255.0

Command	:SYSTem:COMMunicate:LAN:GATEway
Function	Set/query the gateway address of LAN port.
Syntax	:SYSTem:COMMunicate:LAN:GATEway {<Address>}
	:SYSTem:COMMunicate:LAN:GATEway?->{<Address>}
	<Address>= Gateway address in the format "x.x.x.x"
Example	:SYSTem:COMMunicate:LAN:GATEway "192.168.1.1"
	:SYSTem:COMMunicate:LAN:APPLy
	:SYSTem:COMMunicate:LAN:GATEway?->192.168.1.1

Command	:SYSTem:ERRor[:NEXT]?
Function	Query error code in SCPI.
Syntax	:SYSTem:ERRor[:NEXT]?->{<errorNum>,<errorString>}
	<errorNum>= Error code
	<errorString>= Error character string
Example	:SYSTem:ERRor[:NEXT]?
	->0,"No error"

Command	:SYSTem:ERRor:COUNt?
Function	Query error code and character string in SCPI.
Syntax	:SYSTem:ERRor:COUNt?->{<errorCount>}
	<errorCount>= Error queue number

Example	:SYSTem:ERRor:COUNt?
	->0

Command	:SYSTem:VERSion?
Function	Query SCPI version number.
Syntax	:SYSTem:VERSion?->{<Version>}
	<Version>=SCPI version number
Example	:SYSTem:VERSion?
	1999

Command	*IDN?
Function	Query the instrument information.
Syntax	*IDN? ->{<IDN1>,<IDN2>,<IDN3>,<IDN4>}
	<IDN1>: Manufacturer <IDN2>: Instrument model
	<IDN3>: The instrument serial number <software version>
Example	*IDN?
	->Unitrend, UDP5040-40,0000000000000,1.02.0822

STB (Status Byte Register)

The status bit register records the trigger state of other registers. When the register and enable register is non-zero, the bit corresponding to STB will be set. The register is not latched and will change dynamically with the event.

Bit	Decimalism		Definition
0		R01	Not Used
1	2	PRO	Protection Event Flag
2	4	QMA	Error/Event queue message available
3	8	QES	Questionable status
4	16	MAV	Message Available
5	32	ESR	Standard Event Status Register
6	64	SRQ	Service Request
7		OPS	Not Used

*STB

Function Query the status byte event register. The flag bit will automatically be cleared after each read, and the query result will be returned in decimal format.

Syntax *STB?

Example *STB? -> 4

Description If the query returns 4, the status byte register is set to Bit 2, indicating that the error queue is non-zero, which means an error has occurred.

For more details, refer to STB table above.

***SRE**

Function Set the SCPI status byte event enable register.

If the SRE register and STB register are not 0, then the SRQ bit of the STB register will be set to 1.

Syntax *SRE <Value>

*SRE?-><Value>

<Value>= Status byte enable register value

Example *SRE 128

*SRE?->128

ESR (Event Register)

The event register records SCPI and power operation error events. When an event occurs, the register latches the event. The register is cleared only when it is queried or when the *CLS clear command is sent.

Bit	Decimalism		Definition	Description
0		OPC	Operation complete	The default is 0.
1		Not Used		The default is 0.
2	4	QER	Query Error	
3	8	DER	Device Dependent Error	
4	16	EER	Execution Error (e.g. range error)	
5	32	CER	Command error (e.g. syntax error)	
6		Not Used		The default is 0.
7	128	PON	Power On	

Command	*STB?
Function	Query the status byte event register. The flag bit will automatically be cleared after each read, and the query result will be returned in decimal format.
Syntax	*STB?
Example	*STB? -> 4
	If the query returns 4, the status byte register is set to Bit 2, indicating that the error queue is non-zero, which means an error has occurred. For more details, refer to STB table above.

Command	*SRE

Function	Set the SCPI status byte event enable register. If the SRE register and STB register are not 0, then the SRQ bit of the STB register will be set to 1.
Syntax	*SRE <Value>
	*SRE?-><Value>
	<Value>= Status byte enable register value
Example	*SRE 128 Set SRE to Bit 7. When a PON (PowerOn) even occurred, the ESR bit in the STB register will be set.
	*SRE?->128

QUES (Questionable Status Register)

Query the working status of the power supply through the status register, including constant current (CC) mode, constant voltage (CV) mode, overtemperature protection (OTP), overvoltage protection (OVP), overcurrent protection (OCP), and other status changes.

Bit	Decimalism		Definition
0	1	CV	CV Mode
1	2	CC	CC Mode
2	4	FAN Error	FAN Error
3	8	Not Used	Not Used
4	16	OTP	Over Temperature
5	32	PFC_Hot	PFC_Hot
6	64	MOS_Hot	MOS_Hot
7	128	OPP	Over Power Protection
8	256	OSP	Over Sense
9	512	OVP	Over Voltage Protection
10	1024	OCP	Over Current Protection
11	2048	Font OCP	Font Over Current
12	4096	Volt Not Calib	No Voltage CalibPoints
13	8192	Curr Not Calib	No Current CalibPoints

Command	:STATus:QUEStionable[:EVENt]?
Function	Query QUSE register in SCPI. The flag bit will automatically be cleared after each read, and the query result will be returned in decimal format.
Syntax	:STATus:QUEStionable[:EVENt]?
Example	:STATus:QUES? -> 512

Description	If the return value is 512, it means Bit 9 of the SCPI QUES register is set, indicating that an Overvoltage Protection (OVP) event has occurred. For more details, refer to QUES table above.

Command	:STATus:QUEStionable:CONDition?
Function	Query QUSE register in SCPI and return the current state.
Syntax	:STATus:QUEStionable:CONDition?
Example	:STATus:QUES:COND? -> 1
Description	If the return value is 1, it means Bit 0 of the SCPI QUES register is set, indicating that the power supply is operating in CV mode. For more details, refer to QUES table above.

Command	:STATus:QUEStionable:ENABle
Function	Set QUSE enable register in SCPI. If the QUES register and QUES enable register are not 0, then the QUES bit of the STB register will be set to 1.
Syntax	:STATus:QUEStionable:ENABle <Value>
	:STATus:QUEStionable:ENABle?-><Value>
	<Value>= QUES time enable register value
Example	:STATus:QUEStionable:ENABle 512
	:STATus:QUEStionable:ENABle?->512
Description	Set QUES bit of the STB register when an OVP (Bit 9: 512) event is triggered.