Faith费思®

Programmable DC Switching Power Supply

Programming Manual

(*FTP series*)

Version V2.02
2020-11

Preface

This manual contains reference information for programming the FTP series programmable switching DC power supply Unit (PSU, i.e. Power Supply Unit) over the remote interface using the SCPI programming language.

Related Information

The applications in this manual assume that you know how to connect the power supply to the computer. Please refer to the user manual for the specific online method.

Part of the content in the manual is related to specific accessories. If you need some special accessories or the accompanied accessories are not enough to meet your needs, please contact Faithtech Technology sales or after-sales service department.

Announcement

For product latest information, please log on to Faithtech's official website http://www.faithtech.cn for inquiries.

Version History

Date	Version	Amendment
2016-3	1.00	Finish this programming manual
2020-7	2.00	Adjust programming examples
2020-8	2.01	Add WAVE programming examples
2020-11	2.02	Add and adjust compatible commands

Table of Contents

1 COMMUNICATION INTERFACE 5

Introduction 5
Configuration for remote communication 5
Connect RS232 6
Connect LAN 7
Enter remote control mode 7
Other information 7

2 SCPI COMMAND AND STATUS REGISTERS 8

Command introduction 8
Command keyword 9
Root Specifier 9
Command separators 10
Query syntax 10
Command terminator 11
Paramter Format 11
Status system 11
Channel Status Register Group 12
Standard Event Status Register Group 13
Status Byte Register 14
Command version information 15

3 SCPI COMMAND DESCRIPTION 16

IEEE488.2 common commands 16
*CLS 16
*ESE 17
*ESR? 17
*IDN? 18
*OPC 18
*PSC 18
*RST 19
*SRE 19
*STB? 19
*SAV 20
*RCL 20
*WAI 21
CONFigure Command 21
CONFigure:TTL 21
CONFigure:FOLD:BACK 21
CONFigure:FOLD:TIME 21
CONFigure:APG:MODE 22
CONFigure:APG:VOLTage 22

CONFigure:MSSL:ID....22
CONFigure:MSSL:PARSer....22
CONFigure:MSSL:NUMSlv....23
CONFigure:MSSL:CONTrol....23
CONFigure:INHibit....23
CONFigure:PRIor....23
OUTPut Command....24
OUTPut[:STATe]....24
OUTPut:FUNCtion....24
OUTPut:PROTection:VOLTage[:LEVel]....24
OUTPut:PROTection:CURRent[:LEVel]....24
OUTPut:PROTection:POWer[:LEVel]....25
OUTPut:PROTection:CLEar....25
FUNCtion Command....25
FUNCtion....25
SOURce Command....26
SOURce:VOLTage[:LEVel]....26
SOURce:VOLTage:SLEW....26
SOURce:VOLTage:LIMit:HIGH....26
SOURce:VOLTage:LIMit:LOW....26
SOURce:CURRent[:LEVel]....27
SOURce:CURRent:SLEW....27
SOURce:CURRent:LIMit:HIGH....27
SOURce:CURRent:LIMit:LOW....28
SEQuence Command....28
SEQuence:STATus....28
SEQuence:RUN:NUMBer....28
SEQuence:EDIT:NUMBer....28
SEQuence:EDIT:COUNt....29
SEQuence:EDIT:CYCLe....29
SEQuence:EDIT:LINK....29
SEQuence:EDIT:SAVE....29
SEQuence:EDIT:STEP....30
SEQuence:EDIT:VOLTage....30
SEQuence:EDIT:CURRent....30
SEQuence:EDIT:DELay....30
CP Command....31
CP:VOLTage....31
CP:CURRent....31
CP:POWer....31
CP:RESPonse....31
MEASure Command....32
MEASure[:SCALar]:VOLTage[:DC]?....32
MEASure[:SCALar]:CURRent[:DC]?....32

MEASure[:SCALar]:POWer[:DC]? 32
MEASure[:SCALar]:TEMPerature? 32
STATus Command 33
STATus:QUEStionable:CONDition? 33
STATus:QUEStionable[:EVENt]? 33
STATus:QUEStionable:ENABle 33
SYSTem Command 33
SYSTem:ERRor? 33
SYSTem:LOCal 34
SYSTem:VERSion? 34
WAVE Command 34
WAVE:STATus 34
WAVE:RUN:NUMBer 34
WAVE:EDIT:NUMBer 35
WAVE:EDIT:COUNt 35
WAVE:EDIT:LINK 35
WAVE:EDIT:CYCLe 35
WAVE:EDIT:SAVE 36
WAVE:EDIT:STEP 36
WAVE:EDIT:ORDer 36
WAVE:EDIT:VOLTage 36
WAVE:EDIT:VSR 37
WAVE:EDIT:CURRent 37
WAVE:EDIT:CSR 37
WAVE:EDIT:TIME 37
WAVE:EDIT:JUTO 38
WAVE:EDIT:JUCN 38
WAVE:EDIT:DELay 38

4 SCPI ERROR INFORMATION 39

Introduction 39
Check error 39
Command Error 39
Excecution error 41
Query error 41

5 SCPI PROGRAMMING EXAMPLES 41

Static VI Output 41
Query readback parameter 42
Constant Power (CP) Output 42
Edit (SEQ) Sequence File 42
Run (SEQ) Sequence File 43
Edit WAVE File 43
Run WAVE File 44

1 Communication Interface

Introduction

Faithtech FTP series programmable DC switching power supply provides various remote communication interfaces such as RS232 port (standard), LAN (standard), GPIB (optional), etc.. You can connect to the power supply through a dedicated cable with the computer, the computer can control the source.

Table 1-1 Communication Interfaces

Remote Controller	Interface	Explanation
PC	RS232	Serial port
	LAN	Standard ethernet

◎ Caution:

You can only select one communication method at a time. Default is RS232.

Configuration for remote communication

This section describes in detail the configuration method of each communication interface. These configurations can only be set via the front panel keyboard of the power supply. For more detailed configuration introduction, please refer to the user manual. Press the "Menu" key to enter the menu, under the "Set" column, select the "System" item, and press the "Enter" key to enter the system parameter setting interface.

Set	Edit	About	
IP	192.168.1.123		
S-Mask	255.255.255.0		
Baud	9600	Shortcut	Off
Parity	None	Power Save	On
Sound	On	Power Output	Off
Language	English		

Figure 1-1 System Parameter Set

Use the knob or direction keys to move the cursor to the setting item, and press the "Enter" key to enter the parameter editing mode. Enter the number keys and decimal point keys to edit the IP address, and turn the knob to select the baud rate and verification mode. The user presses the "Enter" key to confirm the editing parameters, presses the "Esc" key to exit the editing mode.

The communication configuration information is stored in the instrument internal non-volatile memory, and the configuration will not be affected by shutting down or recalling the preset settings. After changing the communication parameters, the device needs to be restarted for the changes to take effect.

Connect RS232

The serial port is a universal asynchronous serial communication interface that conforms to the RS232 level specification and does not support any flow control. RS232 uses 9600 baud rate by default, which can be set to 4800, 19200, 38400 or 115200bps. The baud rate of the power supply and the computer must be the same. The factory default parity is off (no check). If the parity is enabled, the RS232 interface will use odd or even parity to verify data.

For the RS232 interface, only TxD and RxD signals can transmit data, and the pin signals are described in the following table.

Table 1-2 RS232 PIN signal

Pin NO.	Input/Output	Description
1	---	N.C.
2	Input	RxD
3	Output	TxD
4	---	DSR
5	---	GND
6	---	DTR
7	---	CTS
8	---	RTS
9	---	N.C.

Connect LAN

The FTP has an Ethernet communication interface, adopts UDP communication mode, The default IP address is 192.168.1.123, default subnet mask is 255.255.255.0, and the default port number is 7000. Before starting communication, the user needs to set the IP address and subnet mask, and ensure that the address of the PC and the power supply are in the same network segment, and that there is no duplicate IP address with the power supply in the network segment, otherwise the connection will not be correct.

Enter remote control mode

After the power supply receives any correct SCPI command, it enters the remote control mode.

In the remote control mode: the local keyboard is locked, the key operation is invalid, and the power can only be controlled by programming commands; the front panel screen displays real-time status information such as voltage, current, and power, etc.. There are two ways to exit the remote control mode:

※ Press "Enter" key, the system returns to local operation mode.

※ Send the programming command "SYSTem:LOCal" to make the power supply return to local mode.

Other information

For instructions on related software operations, driver installation and communication operations, please refer to the user manual and the accompanying software instructions. For the latest information about the software and drivers, please log in to Faithtech's website http://www.faithtech.cn for inquiries.

2 SCPI Command And Status Registers

Command introduction

SCPI commands can be divided to common and subsystem commands.

Common commands are defined by the IEEE 488.2 standard to perform common interface functions. They begin with an * and consist of three letters (command) or three letters and a ? (query).

Subsystem commands are specific to instrument functions. They can be a single command or a group of commands. The groups are comprised of commands that extend one or more levels below the root.

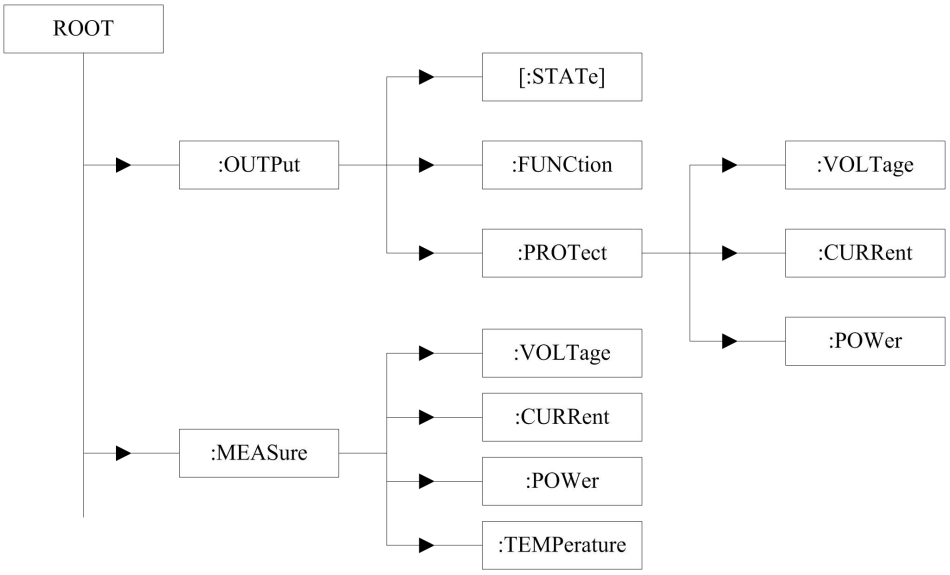

Figure 2-1 Command Levels

Programming Command Syntax

FTP series programmable DC power supply SCPI commands are inherited and expanded from IEEE488.2 standards. SCPI commands are constructed with keyword, seperator, paramter and terminator. Below is an example,

CURRent:STATic:L1 10.0

In this command, CURRent, STATic, L1 are command keywords, ":" and space character are seperator, 10.0 is parameter (some commands have multiple parameters seperated with comma ","), A <carriage return> at the end of this command is the command terminator.

Throughout this document, the following conventions are used for the SCPI command syntax:

- Square brackets ([]) indicate optional keywords or parameters. The braces are not sent with the command string.
- Braces ({}) enclose parameters within a command string.
- Triangle brackets (<>) indicate that you must substitute a value or a code for the enclosed parameter.
- A vertical bar (|) separates one of two or more alternative parameters.

Command keyword

Each command keyword has two formats: long mnemonic and short mnemonic. Short mnemonic is an abbreviation for long mnemonic. Each mnemonic does not exceed 12 characters (including any number suffixes that may appear). The power supply only accepts precise long or short mnemonics. The rules for generating mnemonics are as follows:

The long mnemonic consists of a word or phrase. If it is a word, the entire word constitutes a mnemonic; if it is a phrase, the first character of each word and the entire last word constitute a mnemonic.

CONFIGURE —— CONFigure

Main Value —— MVALue

The short mnemonic is generally composed of the first 4 characters of the long mnemonic.

CONFigure —— CONF

If the character length of the long mnemonic is less than or equal to 4, the long and short mnemonics are the same; if the length of the long mnemonic is greater than 4, and the fourth character is a vowel, the short mnemonic will discard this vowel and becomes 3 characters.

SAVE —— SAVE

TIMer —— TIM

The mnemonic is not case sensitive.

Root Specifier

When it precedes the first header of a message unit, the colon becomes the root specifier. It tells the command parser that this is the root or the top node of the command tree.

Command separators

Colon ":"

A colon (:) is used to separate a command keyword from a lower-level keyword, such as command "CURR:MVAL 10"; also, when it precedes the first header of a message unit, the colon becomes the root specifier, it tells the command parser that this is the root or the top node of the command tree.

Space

Used for seperating command and parameter.

Semicolon ";"

A semicolon (;) is used to separate two commands within the same subsystem, and can also minimize typing. For example, sending the following command string,

"CURR:RANG 0;MVAL 10"

is the same as sending the following two commands:

"CURR:RANG 0"

"CURR:MVAL 10"

Comma ","

A comma "," is used to seperate parameters, as below command,

"CAL:STAT ON,6900"

Query syntax

You can query the value of most parameters by adding a question mark (?) to the command.

For example, the following command sets the output voltage to 80 V:

VOLTage 80

You can query the value by executing:

VOLTage?

After the power supply receives the query command and completes the analysis, it executes the command and generates a response message. The response message is first written into the output buffer. If the current remote interface is a GPIB interface, it will wait for the controller to read the response; otherwise, the response message will be sent to the

interface immediately.

Most of the setting commands have corresponding query syntax. If a command that cannot be queried is received, the power supply will report the error message "-115 Command can not query" and nothing is returned.

Command terminator

There are two types of command terminators: new line character (ASCII symbol LF，ASCII value 10) and EOI (available only in GPIB interface). Command string termination will always reset the current SCPI command path to the root level.

Paramter Format

Table 2-1 Parameter types

Symbol	Explanation	Data Example
<NR1>	Integer value	123
<NR2>	Float value	123.，12.3，0.12，1.23E4
<NRf>	Could be NR1 or NR2。	
<NRf+>	Extended type, including <NRf>, MIN, MAX	
<Bool>	Boolean value	1|0|ON|OFF
<CRD>	String, such as CURR。	
<AARD>	Return ASCII data. Allowed for undefined 7-Bit ASCII. It includes a command terminator.	

Status system

The status system records various conditions and states of the power supply in each status register group. The structure of this status system is shown in the figure below. The status system includes the standard event status register group, the channel status register group, and the status byte register. Each register group consists of multiple registers, including status registers, event registers, and enable registers.

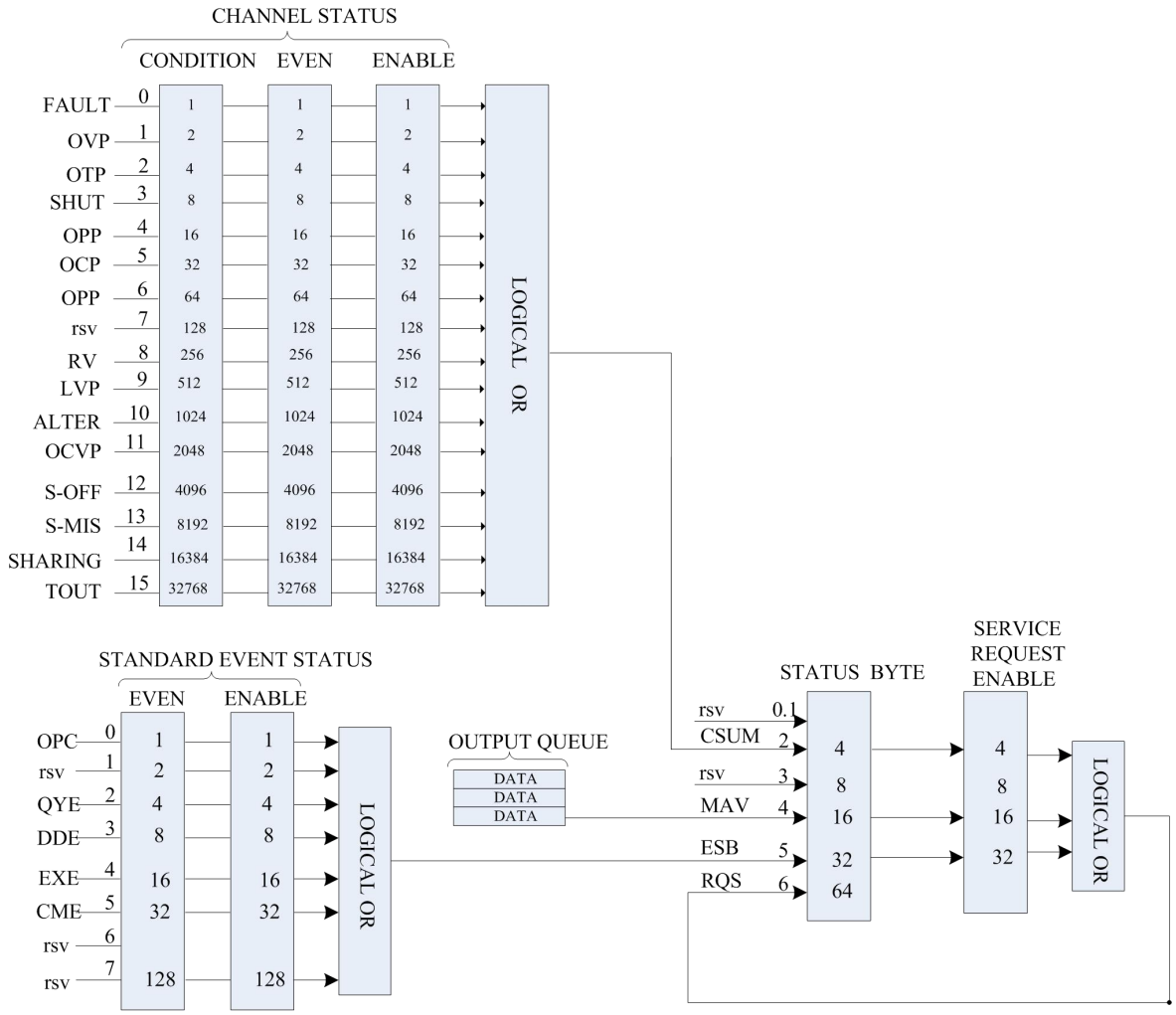

Figure 2-2 Status system of PSU

Channel Status Register Group

The Channel Status Register group reflects the real-time status and events of the power supply, including the Channel Condition register, PTR filter register, NTR filter register, Channel Event register, and Channel Event Enable register.

The Channel Condition register records the real-time status of the power supply. The main content is the alarm information of the power supply, including over-current status, over-voltage status, etc. The detailed definition is shown in the following table.

Table 2-2 Channel Condition Register Bit Explanation

Bit	Name	Explanation
0	FAULT	Module fault
1	OVP	Output voltage exceeds set threshold
2	OTP	Internal temperature too high
3	SHUT	SD signal abnormal
4	OP	Output power exceeds hardware protection threshold
5	OCP	Output current exceeds set threshold
6	OPP	Output power exceeds set threshold
7	rsv	Reserved
8	RV	Output reverse voltage connection
9	LVP	Output voltage below set threshold
10	ALTER	Output mode switched, triggered foldback protection
11	OCVP	Over compensate voltage protection
12	S-OFF	Slave abnormal in master-salve parallel or Serial operation
13	S-MIS	Communication error in master-salve parallel or Serial operation
14	SHARING	Current sharing error in master-salve parallel or Serial operation
15	TOUT	External communication error

The channel event register records the status change event of the power supply, and the meaning of each binary bit corresponds to the bit of the channel condition register. The channel event register can be cleared by the related query command or "*CLS" command. After clearing, it will restart to record new events.

Standard Event Status Register Group

Standard Event Status Register Group records important events that occur during power supply analyzing programming commands or executing operations, including Standard Event registers and Standard Event Enable registers.

The definition of each bit of the standard event register is compatible with the IEEE 488.2 standard, and the detailed definition is as follows:

Table 2-3 Standard Event Register Bit Explanation

Bit	7	6	5	4	3	2	1	0
Name	rsv	rsv	CME	EXE	DDE	QYE	rsv	OPC
OPC All operations & commands completed QYE Query Error DDE Device specific Error EXE Excecution Error CME Command Error								

Bits in the Standard Event register are automatically cleared by a query of that register (such as *ESR?) or by sending the *CLS (clear status) command. Querying an event register returns a decimal value which corresponds to the binary-weighted sum of all bits set in the register. The Standard Event ENABle register is used to define which bits of the Standard Event register will latch ESB (bit 5) of the Status Byte register.

Status Byte Register

The Status Byte Register records important states that IEEE 488.2 bus-compatible devices need to support. Its status bits record whether there are currently unserviced events, errors, standard events, etc. in the power supply.

The bits definition of the Status Byte Register are fully compatible with IEEE 488.2 specifications, details are as follows:

Table 2-4 Status Byte Register Bits

Bit		6	5	4	3	2	1	0
Name		RQS	ESB	MAV	rsv	CSUM	rsv	rsv
CSUM The summary bit for the Channel Status Register group MAV (Message Available) This is set when there is data in the Output Queue ESB The summary bit for the Standard Event Status Register group RQS Exsit Request for service rsv Reserved								

Command version information

The version information of programming commands can only be queried remotely. Implement the following command via remote interface to check PSU's programming command version:

SYSTem:VERSion?

The return message format "YYYY.V", and "YYYY" stands for year, "V" stands for version code.

3 SCPI Command description

IEEE488.2 common commands

This section summarizes the mandatory subset of IEEE 488.2 commands required for any SCPI compliant instrument.

Common command	Description
*CLS	Clears all event registers, besides status byte register and error queue
*ESE {<value>}	Programs bits in the Standard Event Enable register
*ESR?	Query the Standard Event Register
*IDN?	Returns the UNIQUE identification of the PSU
*OPC	Operation Complete Command used for program synchronization
*RCL {<profile>}	Recalls the PSU state stored in the specified storage location
*RST	Reset PSU to the initial state
*SAV {<profile>}	Stores the current PSU state in the specified storage location
*SRE	Programs bits in the Status Byte enable register.
*STB?	Query the Status Byte register
*PSC	Define power on state for service request enable register and standard event enable register
*TST?	Returns Self-Test results
*WAI	Waits until all pending commands are completed

*CLS

Clear command. This command clears all event registers in the PSU:

- ✧ Standard Event Register
- ✧ Channel Event Register
- ✧ Status Byte Register
- ✧ Error Queue

Command syntax: *CLS

Parameter: None

Query syntax: None

*ESE

This command sets the Standard Event Enable register bits in the PSU. A 1 in the bit position enables the corresponding event. All of the enabled events of the Standard Event Enable Register are logically ORed sets the Event Summary Bit (ESB) of the Status Byte Register.

Command syntax: *ESE <NR1>

Parameter: 0～255

The value when power on: refer to *PSC command.

Usage example: *ESE 128

Query syntax: *ESE?

Return: <NR1> (returns a decimal value which corresponds to the binary-weighted sum of all bits in the Standard Event Enable register)

Related commands: *ESR?, *PSC, *STB?

Standard Event Register Bit Explanation

Bit	7	6	5	4	3	2	1	0
Name	rsv	rsv	CME	EXE	DDE	QYE	rsv	OPC
OPC All operations & commands completed QYE Query Error DDE Device specific Error EXE Excecution Error CME Command Error								

*ESR?

Standard Event Register Query. Reading the Standard Event register clears it.

Query syntax: *ESR?

Parameter: None

Return: <NR1> (returns a decimal value which corresponds to the binary-weighted sum of all bits in the Standard Event register)

Related commands: *CLS, *ESE, *OPC

*IDN?

Queries the manufacturer, model name, reserved code, and firmware version of the PSU.

Query syntax: *IDN?

Parameter: None

Return: <AARD>

Return example: Faith, FTP, 0, V1.00

*OPC

It causes the PSU to set the OPC bit (bit 0) of the Standard Event register. OPC bit value is “1”, that is all commands before *OPC operation have been completed.

Command syntax: *OPC

Parameter: None

Query syntax: *OPC?

Return: <NR1>

Related commands: *TRG, *WAI

*PSC

It determines to save or not for the Service Request Enable Register and Standard Event Enable Register at the moment when PSU is powered on.

1: Not save. When powered on, the PSU will clear the Service Request Enable Register and Standard Event Enable Register.

0: Save. When powered on, the PSU can recall and use the previously saved value.

Command syntax: *PSC <bool>

Parameter: 0 | 1

Usage example: *PSC 1

Query syntax: *PSC?

Return: OFF | ON（return *PSC current status）

Related commands: *ESE, *SRE

*RST

Reset Command. Restores the PSU to its initial factory default state

Command syntax: *RST

Parameter: None

Return: None

Related commands: None

*SRE

Sets or queries the Service Request Enable register. The Service Request Enable register determines which bits of the Status Byte register are able to generate service requests.
The RQS bit of the Status Byte Register will be set “1” when the same bit are “1” for both Status Byte Register and Service Request Enable Register.

Command syntax: *SRE <NR1>

Parameter: 0~255

Query syntax: *SRE?

Return: <NR1>

Related commands: *ESE, *ESR, *PSC

*STB?

Query Status Byte register. The Status Byte Register bits are cleared when it is read.

Command syntax: *STB?

Parameter: None

Return value: <NR1>

Relating commands: *CLS *ESE *ESR

*SAV

This command stores the current instrument state in the specified storage location. Any state previously stored in the same location is overwritten without generating any errors. The PSU has 20 storage locations in non-volatile memory which are available to the user for storing PSU states. (location 1 ~ 20)

Command syntax: *SAV <NR1>

Parameter: 1～20

Usage example: *SAV 3

Query syntax: None

Related commands: *RCL

◎ Caution:

*SAV command takes an excution time of 500ms, do not operate the instrument while implementing this command.

*RCL

This command recalls the PSU state stored in the specified storage location. The PSU has 20 storage locations in non-volatile memory to store PSU states. It is not possible to recall the PSU state from a storage location that is empty or was deleted.

Command syntax: *RCL <NR1>

Parameter: 1～20

Usage example: *RCL 3

Query syntax: None

Related commands: *SAV

◎ Caution:

*RCL command takes an excution time of 500ms, do not operate the instrument while implementing this command.

*WAI

The Wait-to-Continue Command causes the PSU to wait until all pending commands are completed before executing any other commands.

Command syntax: *WAI

Parameter: None

Relating commands: *OPC

CONFigure Command

CONFigure:TTL

Set output value for digit signal output terminal.

Command syntax: CONFigure:TTL <NR1>

Parameter: 0～15

Example: CONF:TTL 6

Query syntax: CONFigure:TTL?

Return value: <NR1>

CONFigure:FOLD:BACK

Set foldback protection mode, when power supply switches output mode between CC and CV, the output will be turned off.

Command syntax: CONFigure:FOLD:BACK <NR1>

Parameter: 0 | 1 | 2 | OFF | CV2CC | CC2CV

Example: CONF:FOLD:BACK 1

Query syntax: CONFigure:FOLD:BACK?

Return value: <NR1>

CONFigure:FOLD:TIME

Set foldback protection delay time.

Command syntax: CONFigure:FOLD:TIME <NRf>

Parameter: 0.1～600

Unit: s (second)

Example: CONF:FOLD:TIME 1

Query syntax: CONFigure:FOLD:TIME?

Return value: <NRf>

CONFigure:APG:MODE

Set analog programming mode.

Command syntax: CONFigure:APG:MODE <NR1>

Parameter: 0 | 1 | 2 |3 | OFF | V | I | V&I

Unit: None

Example: CONF:APG:MODE 1

Query syntax: CONFigure:APG:MODE?

Return value: <NR1>

CONFigure:APG:VOLTage

Set analog programming reference voltage.

Command syntax: CONFigure:APG:VOLTage <NR1>

Parameter: 0 | 1 | REF5 | REF10

Unit: None

Example: CONF:APG:VOLT 1

Query syntax: CONFigure:APG:VOLTage?

Return value: <NR1>

CONFigure:MSSL:ID

Set PSU ID in master-slave operation, also used as communication address.

Command syntax: CONFigure:MSSL:ID <NR1>

Parameter: 0 | 1 | 2 | 3 | 4 | MASTER | SLAVE1 | SLAVE2 | SLAVE3 | SLAVE4

Example: CONF:MSSL:ID 0

Query syntax: CONFigure:MSSL:ID?

Return value: MASTER | SLAVE1 | SLAVE2 | SLAVE3 | SLAVE4

CONFigure:MSSL:PARSer

Set power supply as master-slave parallel or series operation.

Command syntax: CONFigure:MSSL:PARSer <NR1>

Parameter: 0 | 1 | PARALLEL | SERIES

Example: CONF:MSSL:PARS 1

Query syntax: CONFigure:MSSL:PARSer?

Return value: PARALLEL | SERIES

CONFigure:MSSL:NUMSlv

Set number of slave units in master-slave operatoin.

Command syntax: CONFigure:MSSL:NUMSlv <NR1>

Parameter: 1～4

Example: CONF:MSSL:NUMS 4

Query syntax: CONFigure:MSSL:NUMSlv?

Return value: <NR1>

CONFigure:MSSL:CONTrol

Enable or disable master-slave control.

Command syntax: CONFigure:MSSL:CONTrol <NR1>

Parameter: 0 | 1 | OFF | ON

Example: CONF:MSSL:CONT 0

Query syntax: CONFigure:MSSL:CONTrol?

Return value: OFF | ON

CONFigure:INHibit

Set external signal control behavior for PSU output.

Command syntax: CONFigure:INHibit <NR1>

Parameter: 0 | 1 | 2 | OFF | TOGGLE | HOLD

Unit: None

Example: CONF:INH 2

Query syntax: CONFigure:INHibit?

Return value: OFF | TOGGLE | HOLD

CONFigure:PRIor

Set constant voltage priority start or constant current priority start.

Command syntax: CONFigure:PRIor <NR1>

Parameter: 0 | 1 | VOLTAGE | CURRENT

Unit: None

Example: CONF:PRI 1

Query syntax: CONFigure:PRIor?

Return value: VOLTAGE | CURRENT

OUTPut Command

OUTPut[:STATe]

Turn on/off output, also used for turn on/off test functin output.

Command syntax: OUTPut[:STATe] <bool>

Parameter: 0 | 1 | OFF | ON

Example: OUTP ON

Query syntax: OUTPut[:STATe]?

Return value: OFF | ON

OUTPut:FUNCtion

Switch power supply test function.

Command syntax: OUTPut:FUNCtion <NR1>

Parameter: 0 | 1 | 2 | 3 | VI | WAVE | CP | SEQ

Example: OUTP:FUNC VI

Query syntax: OUTPut:FUNCtion?

Return value: 0 | 1 | 2 | 3

OUTPut:PROTection:VOLTage[:LEVel]

Set power supply overvoltage protection threshold.

Command syntax: OUTPut:PROTection:VOLTage[:LEVel] <NRf>

Parameter: MIN～MAX

Unit: V (Volt)

Example: OUTP:PROT:VOLT 10

Query syntax: OUTPut:PROTection:VOLTage[:LEVel]?

Return value: <NRf>[Unit=V]

OUTPut:PROTection:CURRent[:LEVel]

Set power supply overcurrent protection threshold.

Command syntax: OUTPut:PROTection:CURRent[:LEVel] <NRf>

Parameter: MIN～MAX

Unit: A (Ampere)

Example: OUTP:PROT:CURR 20

Query syntax: OUTPut:PROTection:CURRent[:LEVel]?

Return value: <NRf>[Unit=A]

OUTPut:PROTection:POWer[:LEVel]

Set power supply overpower protection threshold.

Command syntax: OUTPut:PROTection:POWer[:LEVel] <NRf>

Parameter: MIN～MAX

Unit: W (Watt)

Example: OUTP:PROT:POW 1000

Query syntax: OUTPut:PROTection:POWer[:LEVel]?

Return value: <NRf>[Unit=W]

OUTPut:PROTection:CLEar

Clear the protection state or fault of the power supply.

Command syntax: OUTPut:PROTection:CLEar

Parameter: None

Example: OUTP:PROT:CLE

Query syntax: None

FUNCtion Command

FUNCtion

Switch the power supply test mode.

Command syntax: FUNCtion <NR1>

Parameter: 0 | 1 | 2 | 3 | VI | WAVE | CP | SEQ

Example: FUNC VI

Query syntax: FUNCtion?

Return value: 0 | 1 | 2 | 3

SOURce Command

SOURce:VOLTage[:LEVel]

Set output voltage.

Command syntax: SOURce:VOLTage[:LEVel] <NRf>

Parameter: MIN～MAX

Unit: V (Volt)

Example: SOUR:VOLT 50.0

Query syntax: SOURce:VOLTage[:LEVel]?

Return value: <NRf>[Unit=V]

SOURce:VOLTage:SLEW

Set voltage slew rate.

Command syntax: SOURce:VOLTage:SLEW <NRf>

Parameter: MIN～MAX

Unit: V/S (Volt/Second)

Example: SOUR:VOLT:SLEW 4000.0

Query syntax: SOURce:VOLTage:SLEW?

Return value: <NRf>[Unit=V/S]

SOURce:VOLTage:LIMit:HIGH

Set output voltage upper limit, in order to protect the DUT.

Command syntax: SOURce:VOLTage:LIMit:HIGH <NRf>

Parameter: MIN～MAX

Unit: V (Volt)

Example: SOUR:VOLT:LIM:HIGH 120.0

Query syntax: SOURce:VOLTage:LIMit:HIGH?

Return value: <NRf>[Unit=V]

SOURce:VOLTage:LIMit:LOW

Set output voltage lower limit, in order to protect the DUT.

Command syntax: SOURce:VOLTage:LIMit:LOW <NRf>

Parameter: MIN～MAX

Unit: V (Volt)

Example: SOUR:VOLT:LIM:LOW 10.0

Query syntax: SOURce:VOLTage:LIMit:LOW?

Return value: <NRf>[Unit=V]

SOURce:CURRent[:LEVel]

Set output current.

Command syntax: SOURce:CURRent[:LEVel] <NRf>

Parameter: MIN～MAX

Unit: A (Ampere)

Example: SOUR:CURR 3.0

Query syntax: SOURce:CURRent[:LEVel]?

Return value: <NRf>[Unit=A]

SOURce:CURRent:SLEW

Set current slew rate.

Command syntax: SOURce:CURRent:SLEW <NRf>

Parameter: MIN～MAX

Unit: A/s (Ampere/second)

Example: SOUR:CURR:SLEW 1500.0

Query syntax: SOURce:CURRent:SLEW?

Return value: <NRf>[Unit=A/S]

SOURce:CURRent:LIMit:HIGH

Set output current upper limit, to protect the DUT.

Command syntax: SOURce:CURRent:LIMit:HIGH <NRf>

Parameter: MIN～MAX

Unit: A (Ampere)

Example: SOUR:CURR:LIM:HIGH 3.5

Query syntax: SOURce:CURRent:LIMit:HIGH?

Return value: <NRf>[Unit=A]

SOURce:CURRent:LIMit:LOW

Set output current lower limit, to protect the DUT.

Command syntax: SOURce:CURRent:LIMit:LOW <NRf>

Parameter: MIN～MAX

Unit: A (Ampere)

Example: SOUR:CURR:LIM:LOW 0.1

Query syntax: SOURce:CURRent:LIMit:LOW?

Return value: <NRf>[Unit=A]

SEQuence Command

SEQuence:STATus

Query sequence running status, return current step number and sequence cycle times.

Command syntax: SEQuence:STATus?

Parameter: None

Example: SEQ:STAT?

Return value: <NR1>,<NR1>

SEQuence:RUN:NUMBer

Set the file number in sequnce test mode.

Command syntax: SEQuence:RUN:NUMBer <NR1>

Parameter: 1～20

Example: SEQ:RUN:NUMB 6

Query syntax: SEQuence:RUN:NUMBer?

Return value: <NR1>

SEQuence:EDIT:NUMBer

Set the file number in sequence edit mode.

Command syntax: SEQuence:EDIT:NUMBer <NR1>

Parameter: 1～20

Example: SEQ:EDIT:NUMB 1

Query syntax: SEQuence:EDIT:NUMBer?

Return value: <NR1>

SEQuence:EDIT:COUNt

Set file length for the sequence being edited.

Command syntax: SEQuence:EDIT:COUNt <NR1>

Parameter: 1～20

Example: SEQ:EDIT:COUN 10

Query syntax: SEQuence:EDIT:COUNt?

Return value: <NR1>

SEQuence:EDIT:CYCLe

Set cycle times for the sequence being edited. Set to 0 means infinit loop.

Command syntax: SEQuence:EDIT:CYCLe <NR1>

Parameter: 0～60000

Example: SEQ:EDIT:CYCL 1

Query syntax: SEQuence:EDIT:CYCLe?

Return value: <NR1>

SEQuence:EDIT:LINK

Set linked sequence for the sequence being edited. 0 means no link.

Command syntax: SEQuence:EDIT:LINK <NR1>

Parameter: 0～20

Example: SEQ:EDIT:LINK 0

Query syntax: SEQuence:EDIT:LINK?

Return value: <NR1>

SEQuence:EDIT:SAVE

Save the sequnce file that is being edited.

Command syntax: SEQuence:EDIT:SAVE

Parameter: None

Example: SEQ:EDIT:SAVE

Query syntax: None

SEQuence:EDIT:STEP

Set the step number for current step being edited.

Command syntax: SEQuence:EDIT:STEP <NR1>

Parameter: 1～20

Example: SEQ:EDIT:STEP 1

Query syntax: SEQuence:EDIT:STEP?

Return value: <NR1>

SEQuence:EDIT:VOLTage

Set output voltage for current step in sequence file.

Command syntax: SEQuence:EDIT:VOLTage <NRf>

Parameter: MIN～MAX

Unit: V (Volt)

Example: SEQ:EDIT:VOLT 12.0

Query syntax: SEQuence:EDIT:VOLTage?

Return value: <NRf>[Unit=V]

SEQuence:EDIT:CURRent

Set output current for step in sequence file.

Command syntax: SEQuence:EDIT:CURRent <NRf>

Parameter: MIN～MAX

Unit: A (Ampere)

Example: SEQ:EDIT:CURR 2.0

Query syntax: SEQuence:EDIT:CURRent?

Return value: <NRf>[Unit=A]

SEQuence:EDIT:DELay

Set delay time of the current step in sequence file.

Command syntax: SEQuence:EDIT:DELay <NRf>

Parameter: MIN～MAX

Unit: s (second)

Example: SEQ:EDIT:DEL 1.0

Query syntax: SEQuence:EDIT:DELay?

Return value: <NRf>[Unit=s]

CP Command

CP:VOLTage

Set maximum output voltage in CP output mode.

Command syntax: CP:VOLTage <NRf>

Parameter: MIN～MAX

Unit: V (Volt)

Example: CP:VOLT 500.0

Query syntax: CP:VOLTage?

Return value: <NRf>[Unit=V]

CP:CURRent

Set maximum output current in CP output mode.

Command syntax: CP:CURRent <NRf>

Parameter: MIN～MAX

Unit: A (Ampere)

Example: CP:CURR 20.0

Query syntax: CP:CURRent?

Return value: <NRf>[Unit=A]

CP:POWer

Set maximum output power in CP output mode.

Command syntax:CP:POWer <NRf>

Parameter: MIN～MAX

Unit: W (Watt)

Example: CP:POW 1000.0

Query syntax: CP:POWer?

Return value: <NRf>[Unit=W]

CP:RESPonse

Set response speed in CP output mode.

Command syntax:CP:RESPonse <NR1>

Parameter: 1～100%

Example: CP:RESP 50

Query syntax: CP:RESPonse?

Return value: <NR1>%

MEASure Command

MEASure[:SCALar]:VOLTage[:DC]?

Enquire the actual output voltage.

Command syntax：MEASure[:SCALar]:VOLTage[:DC]?

Parameter: None

Example：MEAS:VOLT?

Return parameter：<NR2>[Unit=V]

MEASure[:SCALar]:CURRent[:DC]?

Enquire the actual output current.

Command syntax: MEASure[:SCALar]:CURRent[:DC]?

Parameter：None

Example：MEAS:CURR?

Return：<NR2>[Unit=A]

MEASure[:SCALar]:POWer[:DC]?

Enquire the actual output power.

Command syntax:MEASure[:SCALar]:POWer[:DC]?

Parameter: None

Example: MEAS:POW?

Return value: <NR2>[Unit=W]

MEASure[:SCALar]:TEMPerature?

Enquire the actual module temperature.

Command syntax: MEASure[:SCALar]:TEMPerature?

Parameter: None

Example: MEAS:TEMP?

Return value: <NR2>[Unit=℃]

STATus Command

STATus:QUEStionable:CONDition?

Query the Channel Condition register.

Command syntax: STATus:QUEStionable:CONDition?

Parameter: None

Example: STAT:QUES:COND?

Return value: <NR1>

STATus:QUEStionable[:EVENt]?

Query the Channel Event register. The Channel Event register is cleared after read.

Command syntax: STATus:QUEStionable[:EVENt]?

Parameter: None

Example: STAT:QUES?

Return value: <NR1>

STATus:QUEStionable:ENABle

Set the Channel Event Enable register.

Command syntax: STATus:QUEStionable:ENABle <NR1>

Parameter: 0～65535

Example: STAT:QUES:ENAB 65535

Query syntax: STATus:QUEStionable:ENABle?

Return value: <NR1>

SYSTem Command

SYSTem:ERRor?

Query the Error Queue.

Command syntax: SYSTem:ERRor?

Parameter: None

Example: SYST:ERR?

Return value: <NR1>,<SRD>

SYSTem:LOCal

Exsit remote control mode, enter local control.

Command syntax: SYSTem:LOCal

Parameter: None

Example: SYST:LOC

Query syntax: None

SYSTem:VERSion?

Query system version information.

Command syntax: SYSTem:VERSion?

Parameter: None

Example: SYST:VERS?

Return value: <NR2>

WAVE Command

WAVE:STATus

Query wave running state, return wave file number and current running step.

Command syntax: WAVE:STATus?

Parameter: None

Example: WAVE:STAT?

Return value: <NR1>,<NR1>

WAVE:RUN:NUMBer

Set the file number in wave running test.

Command syntax: WAVE:RUN:NUMBer <NR1>

Parameter: 1

Example: WAVE:RUN:NUMB 1

Query syntax: WAVE:RUN:NUMBer?

Return value: <NR1>

WAVE:EDIT:NUMBer

Set file number in wave edit.

Command syntax: WAVE:EDIT:NUMBer <NR1>

Parameter: 1

Example: WAVE:EDIT:NUMB 1

Query syntax: WAVE:EDIT:NUMBer?

Return value: <NR1>

WAVE:EDIT:COUNt

Set file length for the wave being edited.

Command syntax: WAVE:EDIT:COUNt <NR1>

Parameter: 1～2048

Example: WAVE:EDIT:COUN 100

Query syntax: WAVE:EDIT:COUNt?

Return value: <NR1>

WAVE:EDIT:LINK

Set link file for the wave being edited. 0 means no link.

Command syntax: WAVE:EDIT:LINK <NR1>

Parameter: 0

Example: WAVE:EDIT:LINK 0

Query syntax: WAVE:EDIT:LINK?

Return value: <NR1>

WAVE:EDIT:CYCLe

Set running cycle times for the wave being edited. 0 means infinit loop.

Command syntax: WAVE:EDIT:CYCLe <NR1>

Parameter: 0～60000

Example: WAVE:EDIT:CYCL 0

Query syntax: WAVE:EDIT:CYCLe?

Return value: <NR1>

WAVE:EDIT:SAVE

Save the wave file which is being edited.
Command syntax: WAVE:EDIT:SAVE
Parameter: None
Example: WAVE:EDIT:SAVE
Query syntax: None

WAVE:EDIT:STEP

Set the number of the step being edited in wave file.
Command syntax: WAVE:EDIT:STEP <NR1>
Parameter: 1～2048
Example: WAVE:EDIT:STEP 5
Query syntax: WAVE:EDIT:STEP?
Return value: <NR1>

WAVE:EDIT:ORDer

Set function of the step being edited in wave file.
Command syntax: WAVE:EDIT:ORDer <NR1>
Parameter: 0 | 1 | 2 | 3 | 4 | 5 | Voltage | Current | CV_Slew |
CC_Slew | Jump_Loop | Delay
Example: WAVE:EDIT:ORD 1
Query syntax: WAVE:EDIT:ORDer?
Return value: Voltage | Current | CV_Slew | CC_Slew | Jump_Loop | Delay

WAVE:EDIT:VOLTage

Set output voltage of the step being edited in wave file.
Command syntax: WAVE:EDIT:VOLTage <NRf>
Parameter: MIN～MAX
Unit: V (Volt)
Example: WAVE:EDIT:VOLT 12.0
Query syntax: WAVE:EDIT:VOLTage?
Return value: <NRf>[Unit=V]

WAVE:EDIT:VSR

Set voltage slew rate of the step being edited in wave file.

Command syntax: WAVE:EDIT:VSR <NRf>

Parameter: MIN～MAX

Unit: V/S (Volt/Second)

Example: WAVE:EDIT:VSR 5000.0

Query syntax: WAVE:EDIT:VSR?

Return value: <NRf>[Unit=V/s]

WAVE:EDIT:CURRent

Set output current of the step being edited in wave file.

Command syntax: WAVE:EDIT:CURRent <NRf>

Parameter: MIN～MAX

Unit: A (Ampere)

Example: WAVE:EDIT:CURR 2.5

Query syntax: WAVE:EDIT:CURRent?

Return value: <NRf>[Unit=A]

WAVE:EDIT:CSR

Set current slew rate of the step being edited in wave file.

Command syntax: WAVE:EDIT:CSR <NRf>

Parameter: MIN～MAX

Unit: A/s (Ampere/second)

Example: WAVE:EDIT:CSR 1000.0

Query syntax: WAVE:EDIT:CSR?

Return value: <NRf>[Unit=A/S]

WAVE:EDIT:TIME

Set delay time (for Voltage or Current function) of the step being edited in wave file.

Command syntax: WAVE:EDIT:TIME <NR1>

Parameter: MIN～MAX

Unit: ms (minisecond)

Example: WAVE:EDIT:TIME 100

Query syntax: WAVE:EDIT:TIME?

Return value: <NR1>[Unit=ms]

WAVE:EDIT:JUTO

Set the destination step number for a jump from current step being edited in wave file.

Command syntax: WAVE:EDIT:JUTO <NR1>

Parameter: 1～2048

Example: WAVE:EDIT:JUTO 5

Query syntax: WAVE:EDIT:JUTO?

Return value: <NR1>

WAVE:EDIT:JUCN

Set the running cycle times (jump to self) for the step being edited in wave file.

Command syntax: WAVE:EDIT:JUCN <NR1>

Parameter: 0～60000

Example: WAVE:EDIT:JUCN 10

Query syntax: WAVE:EDIT:JUCN?

Return value: <NR1>

WAVE:EDIT:DELay

Set delay time of the step being edited in wave file.

Command syntax: WAVE:EDIT:DELay <NR1>

Parameter: 0～86400

Unit: s (second)

Example: WAVE:EDIT:DEL 100

Query syntax: WAVE:EDIT:DELay?

Return value: <NR1>[Unit=s]

4 SCPI Error Information

Introduction

Any errors that occur during the work process are recorded in the error queue until the error queue is full. The error information can be read through the panel menu or programming commands.

Errors are retrieved in the order of first-in, first-out, and the first error returned is the earliest error. Each time it is read, one error item is deleted from the error queue. If there is no error currently, i.e. the error queue is empty, the power supply will return the message "+0 No error" when sending the query command.

Check error

In remote control mode, execute the following command to read and clear an error message in the queue:

SYSTem:ERRor?

The information returned by this command is a string, such as:

"+101 Invalid character"

This error message indicates that there are invalid characters in the command string received by the power supply. If all error information is read or no error occurs when query the error queue, executing the "SYSTem:ERRor?" command will return information:

"+0 No error"

This message means that there is no error or that the error message has all been cleared.

The following subsections will describe in detail the meanings of error codes which is returned by the power supply.

Command Error

-100	Command error
-101	Invalid character
-102	Syntax error
-103	Invalid separator
-104	Data type error

-105	GET not allowed
-106	Semicolon unwanted
-107	Comma unwanted
-108	Parameter not allowed
-109	Missing parameter
-110	Command header error
-111	Header separator error
-112	Program mnemonic too long
-113	Undefined header
-114	Header suffix out of rang
-115	Command can not query
-116	Command must query
-120	Numeric data error
-121	Invalid character in number
-123	Exponent too large
-124	Too many digits
-128	Numeric data not allowed
-130	Suffix error
-131	Invalid suffix
-134	Suffix too long
-138	Suffix not allowed
-140	Character data error
-141	Invalid character data
-144	Character data too long
-148	Character data not allowed
-150	String data error
-151	Invalid string data
-158	String data not allowed
-160	Block data error
-161	Invalid block data
-168	Block data not allowed
-170	Expression error
-171	Invalid expression
-178	Expression data not allowed
-180	Macro error
-181	Invalid outside macro definition
-183	Invalid inside macro definition
-184	Macro parameter error

Excecution error

-200	Execution error
-220	Parameter error
-221	Setting conflict
-222	Data out of range
-224	Illegal paramter value
-225	Out of memory
-232	Invalid format
-240	Hardware error
-242	Calibration data lost
-243	NO reference
-256	File name not found
-259	Not selected file
-295	Input buffer overflow
-296	Output buffer overflow

Query error

-350	Query overflow
-400	Query error

5 SCPI Programming Examples

Static VI Output

```
OUTP OFF                    //Must turn off output before switching output mode
OUTP:FUNC VI                //Switch to static VI output
SOUR:VOLT:SLEW 5000         //Set voltage slew rate as 5000V/s
SOUR:CURR:SLEW 2000         //Set current slew rate as 2000A/s
SOUR:VOLT 10                //Set output voltage as 10V
SOUR:CURR 10                //Set output current as 10A
OUTP ON                     //Turn on output
SOUR:VOLT 20                //Set output voltage as 20V
```

Query readback parameter

```
MEAS:VOLT?                  //enquire the actual output voltage
MEAS:CURR?                  //enquire the actual output current
MEAS:POW?                   //enquire the actual output power
MEAS:VOLT?;CURR?;POW ?      //enquire the actual output voltage, current, power
```

Constant Power (CP) Output

```
OUTP OFF                    //Must turn off output before CP output
OUTP:FUNC CP                //Switch to CP output
CP:VOLT 100.0               //Set maximum voltage as 100V
CP:CURR 3.0                 //Set maximum current as 3A
CP:POW 100.0                //Set constant output power at 100W
CP:RESP 100                 //Set CP response speed 100%
OUTP ON                     //Turn on output
CP:POW 200.0                //Change constant output power to 200W
```

Edit (SEQ) Sequence File

```
OUTP OFF                    //Must turn off output before sequence editing
SEQ:EDIT:NUMB 1             //Edit sequence file, file number 1
SEQ:EDIT:COUN 3             //Set file length as 3 steps
SEQ:EDIT:CYCL 1             //Set sequence file to be run for once only
SEQ:EDIT:LINK 0             //Set link to file number, 0 means no linked file.
SEQ:EDIT:STEP 1             //Edit the first step
SEQ:EDIT:VOLT 12.0          //Set output voltage as 12V
SEQ:EDIT:CURR 1.0           //Set output current as 1A
SEQ:EDIT:DEL 1.0            //Set step delay time as 1s
SEQ:EDIT:STEP 2             //Edit the second step
SEQ:EDIT:VOLT 24.0          //Set output voltage as 24V
SEQ:EDIT:CURR 2.0           //Set output current as 2A
SEQ:EDIT:DEL 2.0            //Set step delay time as 2s
SEQ:EDIT:STEP 3             //Edit the third step
SEQ:EDIT:VOLT 36.0          //Set output voltage as 36V
SEQ:EDIT:CURR 3.0           //Set output current as 3A
```

```
SEQ:EDIT:DEL 3.0          //Set step delay time as 3s
SEQ:EDIT:SAVE             //Save the sequence file
```

Run (SEQ) Sequence File

```
OUTP OFF                  //Must turn off output before switching test output
OUTP:FUNC 3               //Switch to SEQ test output
SEQ:RUN:NUMB 1            //Select sequence, file number 1
OUTP ON                   //Turn on output
```

Edit WAVE File

```
OUTP OFF                  //Must turn off output before wave editing
WAVE:EDIT:NUMB 1          //Edit WAVE file, file number is 1
WAVE:EDIT:COUN 7          //Set WAVE file length is 7
WAVE:EDIT:LINK 0          //Set as no linked wave file
WAVE:EDIT:CYCL 10         //Set the wave file to be run for 10 times
WAVE:EDIT:STEP 1          //Edit the first step of the wave file
WAVE:EDIT:ORD voltage     //Set step output constant voltage
WAVE:EDIT:VOLT 13.5       //Set output voltage as 13.5V
WAVE:EDIT:TIME 20000      //Set delay time as 20000ms
WAVE:EDIT:STEP 2          //Edit the second step of the wave file
WAVE:EDIT:ORD cv_slew     //Set step voltage slew rate
WAVE:EDIT:VSR 1400        //Set voltage slew rate as 1400V/s
WAVE:EDIT:STEP 3          //Edit the third step of the wave file
WAVE:EDIT:ORD voltage     //Set step output constant voltage
WAVE:EDIT:VOLT 6.5        //Set output voltage as 6.5V
WAVE:EDIT:TIME 15         //Set delay time as 15ms
WAVE:EDIT:STEP 4          //Edit the fourth step of the wave file
WAVE:EDIT:ORD cv_slew     //Set step voltage slew rate
WAVE:EDIT:VSR 40          //Set voltage slew rate as 40V/s
WAVE:EDIT:STEP 5          //Edit the fifth step of the wave file
WAVE:EDIT:ORD voltage     //Set step output constant voltage
WAVE:EDIT:VOLT 8.5        //Set output voltage as 8.5V
WAVE:EDIT:TIME 10000      //Set delay time as 10000ms
WAVE:EDIT:STEP 6          //Edit the sixth step of the wave file
WAVE:EDIT:ORD cv_slew     //Set step voltage slew rate
WAVE:EDIT:VSR 50          //Set voltage slew rate as 50V/s
```

```
WAVE:EDIT:STEP 7          //Edit the seventh step of the wave file
WAVE:EDIT:ORD voltage     //Set step output constant voltage
WAVE:EDIT:VOLT 13.5       //Set output voltage as 13.5V
WAVE:EDIT:TIME 2000       //Set delay time as 2000ms
```

Run WAVE File

```
OUTP OFF                  //Must turn off output before switching output mode
OUTP:FUNC 1               //Switch to WAVE output mode
WAVE:RUN:NUMB 1           //Select wave file number 1
OUTP ON                   //Turn on output
```