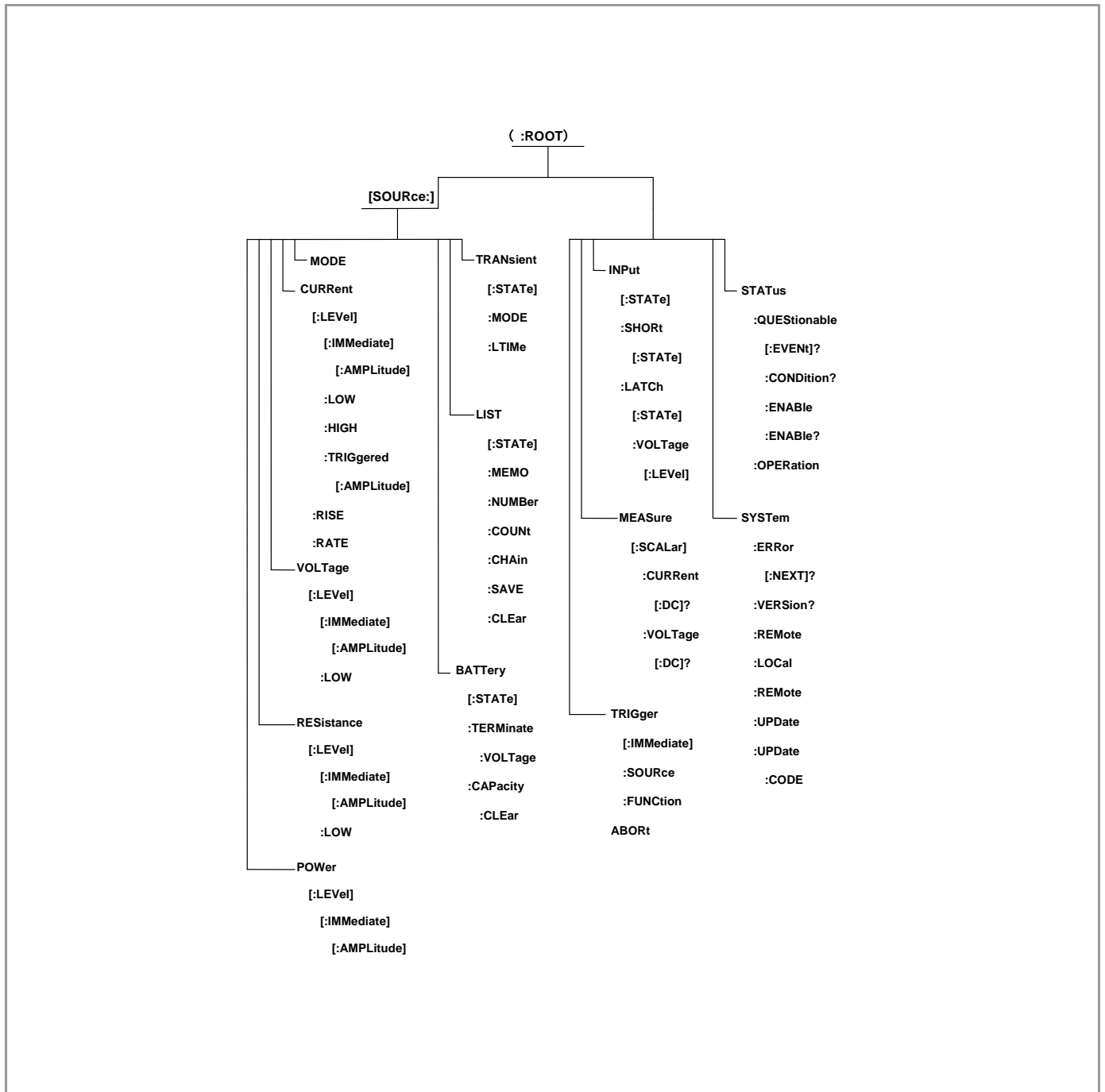


# KONSTANTER SPL

## SPL Series Electronic Load SCPI Programming Guide

3-349-710-03  
3/12.20



Content	Page
<b>1 SCPI Introduction .....</b>	<b>2</b>
1.1 What is SCPI? .....	2
1.2 Command Syntax .....	2
1.2.1 SCPI Keywords .....	2
1.2.1.1 Forms of Keywords .....	2
1.2.1.2 Keyword Conventions .....	3
1.2.2 Separators .....	3
1.2.3 Parameters .....	3
1.2.3.1 Parameter Formats .....	3
1.2.3.2 Data Unit .....	3
1.2.4 Terminators .....	3
1.3 Examples .....	3
1.3.1 Simple Command Statements .....	3
1.3.2 Compound Command Statements .....	3
1.3.3 Simple Command Queries .....	3
1.3.4 Compound Command Queries .....	3
<b>2 SCPI Commands Descriptions .....</b>	<b>4</b>
2.1 IEEE488.2 Common Commands .....	4
2.2 SCPI Root Commands .....	6
2.2.1 ABORt .....	6
2.2.2 CURRent Subsystem .....	6
2.2.3 VOLTage Subsystem .....	7
2.2.4 RESistance Subsystem .....	8
2.2.5 POWer Subsystem .....	8
2.2.6 INPut Subsystem .....	8
2.2.7 TRANsient Subsystem .....	9
2.2.8 LIST Subsystem .....	10
2.2.9 MODE .....	10
2.2.10 MEASure .....	11
2.2.11 STATus Subsystem .....	11
2.2.12 SYSTem .....	12
2.2.13 TRIGger Subsystem .....	12
2.2.14 ATTery Subsystem .....	13
<b>3 Error Messages .....</b>	<b>13</b>
<b>4 Status Register Reporting .....</b>	<b>14</b>
<b>5 Product Support .....</b>	<b>16</b>

## 1 SCPI Introduction

### 1.1 What is SCPI?

SCPI is the Standard Commands for Programmable Instruments based on the IEEE488.1 and IEEE488.2 Standards. It conforms to IEEE754 Standard for Floating-Point Arithmetic, ISO646 7-bit Coded Character Sets for Information Exchange, and other kinds of standards.

SCPI Commands for SPL Series Electronic Load can be divided into two types: one is IEEE488.2 common commands; the other is specific commands for the load.

Some generic commands and queries for instruments are defined by the IEEE488.2 common commands. All IEEE488.2 common commands consist of "\*" and the mnemonic symbol of three characters. (For example: \*IDN?, \*SAV, \*WAI etc.)

The specific commands are used to execute settings, testing, and measuring etc. functions of the load. These commands are organized in the form of the command tree. Each command contains several mnemonic symbols. The nodes are separated by the character ":", see figure 1-1. The top part of the command tree is called "ROOT". The entire path from "ROOT" to leaf node is a complete programming command.

### 1.2 Command Syntax

SCPI Commands for SPL Series Electronic Load carry on and expand IEEE488.2 Standards. The programming command consists of keywords, separators, parameters, and terminators etc. Take the following command as an example:

```
CURRent:PROTection:STATe ON
```

CURRent, PROTection, STATe are the keywords; ":" and spaces are separators; ON is the parameter (some commands have several parameters; they are separated by ";"); the carriage return after the command is a terminator.

To make it easy to describe, the various follow-up symbols represent:

Square brackets ([]) represent the optional keywords or the parameter can be omitted.

Braces ({} ) represent the parameter options in the command string.

Angle brackets (<>) represent that it is necessary to provide a numeric parameter.

Vertical line (|) is used to separate multiple parameter options.

#### 1.2.1 SCPI Keywords

##### 1.2.1.1 Forms of Keywords

Each keyword has two forms: long form and short form.

**Long Form** It consists of one word or a phrase. If it is a word, the long form keyword should be the complete word; if it is a phrase, the long form keyword should consist of the first letter of each word and the whole letters of the last word. The long form keyword for STATUS is STATus; the long form keyword for LOW LEVEL is LLEVEL.

**Short Form** The word contains only the first three or four letters of the long form. The short form keyword for STATUS is STAT; the short form keyword for LOW LEVEL is LLEV.

Short forms are constructed according to the following rules:

- If the keyword consists of four or fewer letters, then all the letters are used.
- If the keyword consists of five or more letters, and the fourth letter is not a vowel, then the first four letters are used; and the fourth letter is a vowel, then only the first three letters are used.

### 1.2.1.2 Keyword Conventions

The short form part of each keyword is emphasized in boldface uppercase letters to help you remember it.

- TRIGger
- IMMEDIATE
- RESistance
- SHORT

The SCPI parser is not sensitive to case. It will accept Trig, trig, trigger, TRIGGER, etc. Regardless of which form you use, the load only recognizes the accurate long form and short form. For example, RESI or TRI will not be recognized as the right command.

### 1.2.2 Separators

The colon “:” is used to separate the keywords of two adjacent levels in command tree. For example: “INPut:PROTEction:CLEAr”. The colon “.” can be regarded as the first character of the command to indicate this command is a root command.

The space “ ” is used to separate the command and the parameter.

The semicolon is used to separate multiple commands without changing the level of the present commands. For example: CURR:PROT:LEV 3;DEL 10. This command is equivalent of the following two commands:

CURR:PROT:LEV 3

CURR:PROT:DEL 10

Using semicolon and colon properly can simplify the description of command level. e.g. CURR 25;VOLT 50. Please note that the maximum length of a command should not exceed 100 bytes. If a command exceeds this length, it will be given up by the load and a remote error “-521 Input buffer overflow” will occur.

The comma “,” is used to separate multiple parameters.

### 1.2.3 Parameters

Parameters are data values or character strings, which follow behind certain keywords, and are separated by spaces from keywords.

#### 1.2.3.1 Parameter Formats

The parameters formats have numerical data format, character data format and Bool format etc. The character format and Boolean format are similar to the keywords format, such as: ON, OFF and CONTinuos. All parameters are ASCII. See the list 1-1 for Parameter Formats.

Symbol	Description	Example
< NR1 >	Digits with no decimal point. The decimal point is assumed to be to the right of the least-significant digit.	2730, 02730
< NR2 >	Digits with a decimal point.	27.30, .02730
< NR3 >	Digits with a decimal point and an exponent.	2.730 E+2, 2.730E-2
< NRf >	Flexible decimal form that includes NR1 or NR2 or NR3.	2730, 27.30, 2.730E+2
< NRf+ >	Expanded decimal form that includes NRf and NRf and MIN, MAX. MIN and MAX are the minimum and maximum limit values for the parameter.	2730, 27.30, 2.730E-2, MIN, MAX
< Bool >	Bool data	ON/OFF
< crd >	Character data	CV
< aard >	Return ASCII data. It's allowed to return undelimited 7-bit ASCII. This data type is an implied terminator.	

List1-1 Parameter Formats

### 1.2.3.2 Data Unit

Numeric data may be followed by the unit. When no unit follows the numeric data, the data unit is the standard unit of this command. See list1-2 for Data Unit, and see list 1-3 for Most-Used Unit Multipliers.

Class	Preferred Unit	Secondary Unit	Referenced Unit
Current	A		Ampere
Resistance	OHM		Ohm
		MOHM	Megohm
Time	s		Second
Amplitude	v		Volt
Power	w		Watt

List1-2 Data Unit

Multiplier	Mnemonic	Definition
1E6	MA	Mega
1E3	K	Kilo
1E-3	M	milli
1E-6	U	micro
1E-9	N	nano

List1-3 Most-Used Unit Multipliers

### 1.2.4 Terminators

The command string sent to the load must be terminated by a <new line> symbol (carriage return character). IEEE-488 EOI (End Or Identify) can be regarded as <new line> symbol to be the terminator of the command. It is acceptable if there is a <new line> symbol after the <carriage return character>.

## 1.3 Examples

### 1.3.1 Simple Command Statements

The simple command statement consists of a command, or keyword, usually followed by a parameter:

VOLT 25

CURR 50

TRIG

### 1.3.2 Compound Command Statements

When two or more keywords are connected by colons, it creates a compound command statement. Usually, the last keyword is followed by a parameter, and is separated by a space:

VOLT:SLEW 1000

CURR:RANG 6

TRIG:SOUR BUS

If the compound command statement contains “[,;]”, the keyword of this part is dispensable.

CURRent:LEVel:TRIGgered 0.5

CURRent:TRIGgered 0.5

These two commands are the same.

### 1.3.3 Simple Command Queries

The simple command query consists of a keyword followed by a question mark (?):

VOLT?

CURR?

### 1.3.4 Compound Command Queries

When two or more keywords are connected by colons and followed by a question mark (?), it creates a compound query statement.

VOLT:TRIG?

CURR:PROT?

MEAS:POW?

## 2 SCPI Commands Descriptions

### 2.1 IEEE488.2 Common Commands

Common commands are defined by the IEEE488.2 standard to perform some of the basic instrument functions, such as recalling, saving, and reset etc. IEEE488.2 Common Commands have no hierarchical structure.

#### \*CLS

Clears the following registers:

- Standard Event register
- Questionable Status Event register
- Operation Status Event register
- Status Byte Event register
- Clears the Error Queue

Command Syntax: \*CLS

Parameters: None

#### \*ESE

This command sets the condition of the Standard Event Status Enable register. A "1" in the bit position enables the corresponding event and report this even to the Status Byte register. Refer to the description about the Standard Event Status Enable register in *Status Register Reporting*

Command Syntax: \*ESE <NRf>

Parameter: 0~255

Power-on Value: refers to \*PSC command

Example: \*ESE 100

Query Syntax: \*ESE?

Returned Parameter: <NR1> 0-255

Related Commands: \*PSC \*STB?

#### \*ESR?

This query reads the Standard Event register. Reading the register clears it. See the list below for the bit configuration of this register.

List 2-1 Bit Definition of Standard Event Register

Bit	7	6	5	4	3	2	1	0
Bit Name	PON	N.U	CME	EXE	DDE	QYE	N.U	OPC
	PON Power On. This bit is set when the load is turned on.			DDE Device Dependent Error.				
	N.U Null, Reserved Bit.			QYE Query Error. When an error occurs to the query, QYE bit will be set.				
	CME Command Error.			N.U Null, Reserved Bit.				
	EXE Execution Error.			OPC Operation Complete.				

Query Syntax: \*ESR?

Parameters: None

Returned Parameter: <NR1>

Related Commands: \*CLS \*OPC

#### \*IDN?

Query the relevant information of the load.

Query Syntax: \*IDN?

Parameters: None

Returned Parameter: <aard>

#### \*OPC

This command causes Bit 0 of the Standard Event Status Register to be set "1" when the load has completed all pending operations.

\*OPC does not prevent processing of subsequent commands. All pending operations are complete including all pending trigger levels have triggered.

Command Syntax: \*OPC

Parameters: None

Related Commands: \*WAI \*OPC?

#### \*OPC?

This query causes the load to place an ASCII "1" in the Output Queue when all pending operations are completed. Unlike \*OPC, \*OPC? prevents processing of all subsequent commands. If there are triggers, and the trigger source has not been previously set to EXT at this time, in this case, the only programmable way to restore operation is by sending the load a GPIB **DCL** (Device Clear) command.

Query Syntax: \*OPC?

Returned Parameter: <NR1>

Related Commands: \*OPC TRIG:SOUR \*WAI

#### \*PSC

This command controls the request service at power turn-on.

1: When the load is power on, the settings of Status Byte Enable register and Standard Event Enable register are cleared to be 0.

0: The present settings of Status Byte Enable register and Standard Event Enable register are saved in nonvolatile memory. The next time the load is turned on, the Standard Event Enable register will be programmed according to the saved settings.

Command Syntax: \*PSC <bool>

Parameter: 0 | 1

Query Syntax: \*PSC?

Returned Parameter: 0 | 1

Related Commands: \*SRE \*ESE

#### \*RCL

This command recalls the saved parameters of the load. The recalled parameters need to be previously stored in memory. When the saved parameters are recalling, the specified location of these parameters must be used. \*RCL also does the following:

- 1 Force an **ABORt** command before resetting any parameters. (This removes all pending trigger levels.)
- 2 After all parameter have been recalled, executes an **INP:PROT:CLE** to clear the load's protection status. At power turn-on, the equivalent of an **\*RCL 0** is executed. Recalling the unsaved location is defaulted to recall the location 0.

Command Syntax: \*RCL <NR1>

Parameter: 0~9

Example: \*RCL 5

Related Commands: \*RST \*SAV

#### \*RST

This command sets the load to its factor-defined state. Force an **ABORt** command before resetting any parameters. After all parameters have been reset, executes an **INP:PROT:CLE** to clear the load's protection status.

Command Syntax: \*RST

Parameter: None

Related Commands: \*RCL \*SAV

### **\*SAV**

This command saves the parameters of the load in the nonvolatile memory. There are 10 storage space in total (locations 0 through 9).

Command Syntax: \*SAV <NR1>

Parameter: 0~9

Example: \*SAV 5

Related Commands: \*RCL \*RST

### **\*SRE**

This command sets the condition of the Status Byte Enable register. The programming parameters determine whether or not a 0 to 1 transition in the Status Byte register will set the RQS bit of the Status Byte register. See *Status Reporting* for more details concerning the Status Byte register.

Command Syntax: \*SRE <NR1>

Parameter: 0~255

Query Syntax: \*SRE?

Returned Parameter: <NR1>

Related Commands: \*PSC

### **\*STB?**

This query reads the Status Byte register. The Status Byte register is cleared when this query has been executed. The \*STB? query and a serial poll share the same returned parameter, but the service request bit (bit 6) of the Status Byte register will not be cleared when a serial poll is sent. See *Status Reporting* for more details concerning the Status Byte register.

Query Syntax: \*STB?

Parameter: None

Returned Parameter: <NR1>

### **\*TRG**

This command which is essentially the same as the Group Execute Trigger (<GET>), generates a trigger to the load only if TRIG:SOUR is set to BUS. This command becomes effective only when the trigger source is set to BUS.

Command Syntax: \*TRG

Parameter: None

Related Commands: TRIG TRIG:SOUR

### **\*TST?**

This query causes the load to go through a limited self-test. The testing does not alter the mode or parameter settings of the load.

Command Syntax: \*TST?

Returned Parameter: <NR1> 0 = test passed    None0= a self-test failure

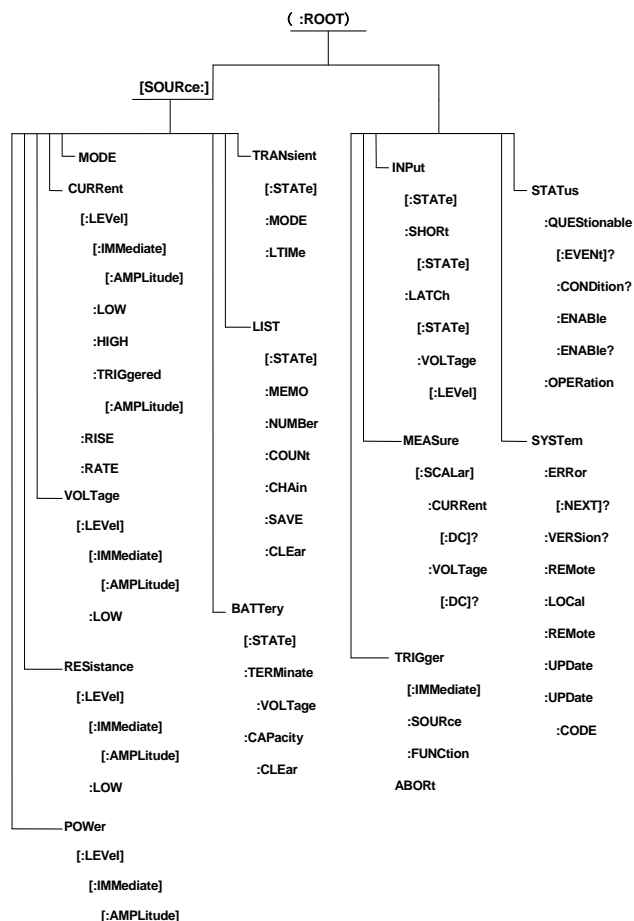
### **\*WAI**

This command instructs the load not process any further commands until all pending operations are completed. All pending operations are complete including all pending trigger levels have triggered. \*WAI can be aborted only by sending the load a GPIB **DCL** (Device Clear) command.

Parameter: None

Related Commands: \*OPC \*OPC?

## 2.2 SCPI Root Commands



### [SOURCE:]CURRENT[:LEVEl]:TRIGgered [:AMPLitude]

Specifies the current trigger level for the load. It shares the same current range with the immediate current. The current trigger level becomes the immediate current level only when the trigger signal has been received.

Command Syntax: [SOURCE:]CURRENT[:LEVEl]:TRIGgered [:AMPLitude] <NRf+>

Parameter: Digits |MIN|MAX

Unit: A|mA

Example: CURR:TRIG 5A CURR:TRIG 50mA

Query Syntax: [SOURCE:]CURRENT[:LEVEl]:TRIGgered [:AMPLitude]?

Parameter: None |MIN|MAX

Example: CURR:TRIG? CURR:TRIG? MIN CURR:TRIG? MAX

Returned Parameter: <NR3>

Related Commands: INIT, INIT:CONT

### [SOURCE:]CURRENT:PROTECTION[:LEVEl]

Specifies the current limit at which software protection occurs. The current limit range is 0-30 A respective 0-40 A. When the current protection is enabled, the actual current exceeds or equals to the current limit for software protection, and the load enters into the current protection status. When the input current reaches the current limit for the specified delay period, the load input is turned off.

Command Syntax: [SOURCE:]CURRENT:PROTECTION[:LEVEl] <NRf+>

Parameter: Digits |MIN|MAX

Unit: A|mA

Example: CURR:PROT 15A

Query Syntax: [SOURCE:]CURRENT:PROTECTION[:LEVEl]?

Parameter: None |MIN|MAX

Example: CURR:PROT? CURR:PROT? MIN CURR:PROT? MAX

Returned Parameter: <NR3>

Related Commands: CURR:PROT:STAT, CURR:PROT:DEL, INP:PROT:CLE

### 2.2.1 ABORT

This command cancels all pending [:LEVEl]:TRIG operations (such as CURR:TRIG). As a result, subsequent triggers have no effect on the input level. ABORT has no effect on the Transient operation and List (Sequence) operation.

Command Syntax: ABORT

Parameter: None

Example: ABOR

Query Syntax: None

Related Commands: CURR[:LEVEl]:TRIG, VOLT[:LEVEl]:TRIG, RES[:LEVEl]:TRIG, STAT:OPER:COND?

### 2.2.2 CURRENT Subsystem

#### [SOURCE:]CURRENT[:LEVEl] [:IMMEDIATE] [:AMPLITUDE]

Specifies the immediate current level for the load. CURR? MIN and CURR? MAX return the maximum and minimum current level for the present range. If the set level exceeds the range, an error will be generated.

Command Syntax: [SOURCE:]CURRENT[:LEVEl] [:IMMEDIATE] <NRf+>

Parameters: Digits |MIN|MAX

Unit: A|mA

Example: CURR 25A

Query Syntax: [SOURCE:]CURRENT[:LEVEl] [:IMMEDIATE] [:AMPLITUDE]?

Parameter: None |MIN|MAX

Example: CURR? CURR? MIN CURR? MAX

Returned Parameter: <NR3>

Related Commands: CURR:LOW; CURR:TRIG, CURR:RISE:RATE

#### [SOURCE:]CURRENT:PROTECTION:DELAY

Specifies time for which protection current limit may be exceeded. The delay period is calculated after the actual current reaches or exceeds the protection current limit. When the delay period is reached, the load input will be turned off.

Command Syntax: CURRENT:PROTECTION:DELAY <NRf+>

Parameter: Digits |MIN|MAX

Unit: s|ms

Example: CURRENT:PROTECTION:DELAY 0.5

Query Syntax: [SOURCE:]CURRENT:PROTECTION:DELAY?

Parameter: None |MIN|MAX

Example: CURR:PROT:DEL? ; CURR:PROT:DEL?MIN ; CURR:PROT:DEL? MAX

Returned Parameter: <NR3>

Related Commands: INP:PROT:CLE, CURR:PROT

#### [SOURCE:]CURRENT:PROTECTION:STATE

Disables/Enables current protection function.

Command Syntax: [SOURCE:]CURRENT:PROTECTION:STATE <bool>

Parameter: ON|OFF

Example: CURR:PROT:STAT ON

Query Syntax: [SOURCE:]CURRENT:PROTECTION:STATE?

Parameter: None

Example: CURR:PROT:STAT?

Returned Parameter: <NR1> Value 0=OFF, 1=ON

Related Commands: CURR:PROT, INP:PROT:CLE

### **[SOURCE:]CURRENT:[LEVEL]:LOW**

Specifies the transient current low level, which shares the same range with the immediate current level.

Command Syntax: [SOURCE:]CURRENT:[LEVEL]:LOW <NRf+>

Parameter: Digits |MIN|MAX

Unit: A|mA

Example: CURR:LOW 3

Query Syntax: [SOURCE:]CURRENT:[LEVEL]:LOW?

Parameter: None |MIN|MAX

Example: CURR:[LEVEL]:LOW? CURR:[LEVEL]:LOW? MIN  
CURR:[LEVEL]:LOW? MAX

Returned Parameter: <NR3>

Related Commands: CURR:LOW

### **[SOURCE:]CURRENT:[LEVEL]:HIGH**

Specifies the transient current high level, which shares the same range with the immediate current level.

Command Syntax: [SOURCE:]CURRENT:[LEVEL]:HIGH <NRf+>

Parameter: Digits |MIN|MAX

Unit: A|mA

Example: CURR:HIGH 3

Query Syntax: [SOURCE:]CURRENT:[LEVEL]:HIGH?

Parameter: None |MIN|MAX

Example: CURR:[LEVEL]:HIGH? CURR:[LEVEL]:HIGH? MIN  
CURR:[LEVEL]:HIGH? MAX

Returned Parameter: <NR3>

Related Commands: CURR:HIGH

### **[SOURCE:]CURRENT:RISE:RATE**

Specifies the current rise rate in CC mode. CURR:RISE:RATE? MIN and CURR:RISE:RATE? MAX return the maximum and minimum rise rate for the present range.

Command Syntax: [SOURCE:]CURRENT:RISE:RATE <NRf+>

Parameter: Digits |MIN|MAX

Unit: A|mA/us

Example: CURR:RISE:RATE 3A

Query Syntax: [SOURCE:]CURRENT:RISE:RATE?

Parameter: None |MIN|MAX

Example: CURR:RISE:RATE? CURR:RISE:RATE? MIN  
CURR:RISE:RATE? MAX

Returned Parameter: <NR3>

Related Commands: CURR:FALL:RATE

### **[SOURCE:]CURRENT:FALL:RATE**

Specifies the current fall rate in CC mode, which shares the same range with the current rise rate.

Command Syntax: [SOURCE:]CURRENT:FALL:RATE <NRf+>

Parameter: Digits |MIN|MAX

Unit: A|mA/us

Example: CURR:FALL:RATE 3A

Query Syntax: [SOURCE:]CURRENT:FALL:RATE?

Parameter: None |MIN|MAX

Example: CURR:FALL:RATE? CURR:FALL:RATE? MIN  
CURR:FALL:RATE? MAX

Returned Parameter: <NR3>

Related Commands: CURR:RISE:RATE

## **2.2.3 VOLTage Subsystem**

### **[SOURCE:]VOLTage:[LEVEL][:IMMEDIATE][:AMPLITUDE]**

Specifies the immediate voltage level for the load. VOLT? MIN and VOLT? MAX return the maximum and minimum voltage level for the present range. If the set level exceeds the range, an error will be generated.

Command Syntax: [SOURCE:]VOLTage:[LEVEL][:IMMEDIATE][:AMPLITUDE] <NRf+>

Parameter: Digits |MIN|MAX

Unit: V|mV

Example: VOLT 5V

Query Syntax: [SOURCE:]VOLTage:[LEVEL][:IMMEDIATE][:AMPLITUDE]?

Parameter: None |MIN|MAX

Example: VOLT? VOLT? MIN VOLT? MAX

Returned Parameter: <NR3>

Related Commands: CURR:ent, RESistance

### **[SOURCE:]VOLTage:[LEVEL]:TRIGGERed[:AMPLITUDE]**

Specifies the voltage trigger level for the load. It shares the same range with the immediate voltage. The voltage trigger level becomes the immediate voltage level only when the trigger signal has been received.

Command Syntax: [SOURCE:]VOLTage:[LEVEL]:TRIGGERed[:AMPLITUDE] <NRf+>

Parameter: Digits |MIN|MAX

Unit: V|mV

Example: VOLT:TRIG 5V

Query Syntax: [SOURCE:]VOLTage:[LEVEL]:TRIGGERed[:AMPLITUDE]?

Parameter: None |MIN|MAX

Example: VOLT:TRIG? VOLT:TRIG? MIN VOLT:TRIG? MAX

Returned Parameter: <NR3>

Related Commands: INIT, INIT:CONT, TRIG

### **[SOURCE:]VOLTage:[LEVEL]:LOW**

Specifies the transient voltage low level, which shares the same range with the immediate voltage level.

Command Syntax: [SOURCE:]VOLTage:[LEVEL]:LOW <NRf+>

Parameter: Digits |MIN|MAX

Unit: V|mV

Example: VOLT:LOW 5V

Query Syntax: [SOURCE:]VOLTage:[LEVEL]:LOW?

Parameter: None |MIN|MAX

Example: VOLT::LOW? VOLT::LOW? MAX VOLT::LOW? MIN

Returned Parameter: <NR3>

Related Commands: VOLT:HIGH, TRAN:LTIM

### **[SOURCE:]VOLTage:[LEVEL]:HIGH**

Specifies the transient voltage high level, which shares the same range with the immediate voltage level.

Command Syntax: [SOURCE:]VOLTage:[LEVEL]:HIGH <NRf+>

Parameter: Digits |MIN|MAX

Unit: V|mV

Example: VOLT:HIGH 5V

Query Syntax: [SOURCE:]VOLTage:[LEVEL]:HIGH?

Parameter: None |MIN|MAX

Example: VOLT:HIGH? VOLT:HIGH? MIN VOLT:[LEVEL]:HIGH? MAX

Returned Parameter: <NR3>

Related Commands: VOLT:LOW, TRAN:HTIM

## 2.2.4 RESistance Subsystem

### [SOURCE:]RESistance[:LEVel][:IMMediate][:AMPLitude]

Specifies the immediate resistance level for the load. RES? MIN and RES? MAX return the maximum and minimum resistance level for the present range. If the set level exceeds the range, an error will be generated.

Command Syntax: [SOURCE:]RESistance[:LEVel][:IMMediate][:AMPLitude] <NRf+>

Parameter: Digits |MIN|MAX

Unit: OHM|mOHM|kOHM

Example: RES 10 OHM

Query Syntax: [SOURCE:]RESistance[:LEVel][:IMMediate]?

Parameter: None |MIN|MAX

Example: RES? RES? MIN RES? MAX

Returned Parameter: <NR3>

Related Commands: CURRent, VOLTage

### [SOURCE:]RESistance[:LEVel]:TRIGgered[:AMPLitude]

Specifies the resistance trigger level for the load. It shares the same range with the immediate resistance level. The resistance trigger level becomes the immediate resistance level only when the trigger signal has been received.

Command Syntax: [SOURCE:]RESistance[:LEVel]:TRIGgered[:AMPLitude] <NRf+>

Parameter: Digits |MIN|MAX

Unit: OHM|mOHM|kOHM

Example: RES:TRIG 3 OHM

Query Syntax: [SOURCE:]RESistance[:LEVel]:TRIGgered[:AMPLitude]?

Parameter: None |MIN|MAX

Example: RES:TRIG? RES:TRIG? MIN RES:TRIG? MAX

Returned Parameter: <NR3>

Related Commands: INIT, INIT:CONT

### [SOURCE:]RESistance[:LEVel]:LOW

Specifies the transient resistance low level, which shares the same range with the immediate resistance level.

Command Syntax: [SOURCE:]RESistance[:LEVel]:LOW <NRf+>

Parameter: Digits |MIN|MAX

Unit: OHM

Example: RES:LOW 3 OHM

Query Syntax: [SOURCE:]RESistance:LOW?

Parameter: None |MIN|MAX

Example: RES:LOW? RES:LOW? MIN RES:LOW? MAX

Returned Parameter: <NR3>

Related Commands: RESistance:HIGH, TRAN:HTIME

### [SOURCE:]RESistance[:LEVel]:HIGH

Specifies the transient voltage high level, which shares the same range with the immediate resistance level.

Command Syntax: [SOURCE:]RESistance[:LEVel]:HIGH <NRf+>

Parameter: Digits |MIN|MAX

Unit: OHM

Example: RES:HIGH 3 OHM

Query Syntax: [SOURCE:]RESistance[:LEVel]:HIGH?

Parameter: None |MIN|MAX

Example: RES:HIGH? RES:HIGH? MIN RES:HIGH? MAX

Returned Parameter: <NR3>

Related Commands: RESistance:HIGH, TRAN:HTIME

## 2.2.5 POWer Subsystem

### [SOURCE:]POWer[:LEVel] [:IMMediate][:AMPLitude]

Specifies the power level. POW? MIN and POW? MAX return the maximum and minimum power level for the present range. If the set level exceeds the range, an error will be generated.

Command Syntax: [SOURCE:]POWer[:LEVel][:IMMediate][:AMPLitude]<NRf+>

Parameter: Digits |MIN|MAX

Unit: W|mW

Example: POW 10W

Query Syntax: [SOURCE:]POWer[:LEVel][:IMMediate][:AMPLitude]?

Parameter: None |MIN|MAX

Example: POW? POW? MIN POW? MAX

Returned Parameter: <NR3>

Related Commands: POWER:TRIGgered

### [SOURCE:]POWer[:LEVel]:TRIGgered[:AMPLitude]

Specifies the power trigger level. It shares the same range with the power level. The power trigger level becomes the power level only when the trigger signal has been received.

Command Syntax: [SOURCE:]POWer[:LEVel]:TRIGgered[:AMPLitude]<NRf+>

Parameter: Digits |MIN|MAX

Unit: W|mW

Example: POW:TRIG 10w

Query Syntax: [SOURCE:]POWer[:LEVel]:TRIGgered[:AMPLitude]?

Parameter: None |MIN|MAX

Example: POW:TRIG? POW:TRIG? MIN POW:TRIG? MAX

Returned Parameter: <NR3>

Related Commands: INIT, INIT:CONT

## 2.2.6 INPut Subsystem

### INPut[:STATe]

Enables/Disables the load input.

Command Syntax: INPut[:STATe] <bool>

Parameter: ON |OFF

Example: INP ON

Query Syntax: INPut[:STATe]?

Parameter: None

Example: INP?

Returned Parameter: <NR1> Value:0(OFF) 1(ON)

Related Commands: INP:SHOR

### INPut:SHORT[:STATe]

Enables/Disables the short circuit of the load.

Command Syntax: INPut:SHORT[:STATe] <bool>

Parameter: ON |OFF

Example: INP:SHOR ON

Query Syntax: INPut:SHORT[:STATe]?

Parameter: None

Example: INP:SHOR?

Returned Parameter: <NR1> Value:0(OFF) 1(ON)

Related Commands: INP[:STAT]



### **INPut:PROTection:CLEar**

Clears the protection status (OC, OV, OP, OT, RV) of the load.

Command Syntax: INPut:PROTection:CLEar

Parameter: None

Example: INP:PROT:CLE

Query Syntax: None

Related Commands: INP

### **INPut:LATCh:VOLTage[:LEVEL]**

Specifies the Von voltage point. If the load is power on, it will be enabled when the external input voltage reaches or exceeds the Von point.

Command Syntax: INPut:LATCh:VOLTage[:LEVEL] <NRf+>

Parameter: Digits |MIN|MAX

Unit: V|mV

Example: INP:LATC:VOLT 2

Query Syntax: INPut:LATCh:VOLTage[:LEVEL]?

Parameter: None |MIN|MAX

Example: INP:LATC:VOLT?; INP:LATC:VOLT?MIN; INP:LATC:VOLT?MAX

Returned Parameter: <NR3>

Related Commands: INP:LATC

### **INPut:LATCh[:STATe]**

Enables/Disables the Von Latch of the load. If the actual voltage is less than Von point after the load has been enabled, there are two circumstances for the load at this time: if the Von Latch is enabled, the load continues operating; if the Von Latch is disabled, the load is turned off.

Command Syntax: INPut:LATCh[:STATe] <bool>

Parameter: ON/OFF

Example: INP:LATC ON

Query Syntax: INPut:LATCh[:STATe]?

Parameter: None

Example: INP:LATC?

Returned Parameter: <NR1> Value:0(OFF) 1(ON)

Related Commands: INP:LATC:VOLT

### **INPut:LIMit[:CV]:CURRent**

Sets the maximum current limit when the load is in the CV mode.

Command Syntax: INPut:LIMit[:CV]:CURRent <NRf+>,

Parameter: Digits |MIN|MAX

Unit: A|mA

Example: INP:LIM:CURR 20

Query Syntax: INPut:LIMit[:CV]:CURRent?

Parameter: None |MIN|MAX

Example: INP:LIM:CURR? INP:LIM:CURR? MIN

INP:LIM:CURR? MAX

Returned Parameter: <NR3>

Related Commands: CURRent

## **2.2.7 TRANSient Subsystem**

### **[SOURce:]TRANSient [:STATe]**

Enables/Disables the transient operation.

Command Syntax: TRANSient[:STATe] <bool>

Parameter: ON|OFF

Example: TRAN ON

Query Syntax: TRAN[:STATe]?

Returned Parameter: <NR1> Value:0 for OFF, 1 for ON

### **TRANSient:MODE**

Sets the transient operating mode. There are three transient operating modes: Continuous, Pulsed, and Toggled.

Command Syntax: TRANSient:MODE <aard>

Parameter: CONTInuous|PULSe|TOGGle

Example: TRAN:MODE PULS

Query Syntax: TRANSient:MODE?

Returned Parameter: <aard>CONT, PULS, or TOGG

Related Commands: TRIGger

### **TRANSient:HTIME**

Sets transient high level time.

Command Syntax: TRANSient:HTIME <NRf+>

Parameter: Digits |MIN|MAX

Unit: mS|uS

Example: TRANSient:HTIME 500ms

Query Syntax: TRANSient:HTIME?

Parameter: None |MIN|MAX

Example: TRAN:HTIME? TRAN:HTIME? MIN TRAN:HTIME? MAX

Returned Parameter: <NR3>

Related Commands: TRAN:LTIM

### **TRANSient:LTIME**

Sets transient low level time.

Command Syntax: TRANSient:LTIME <NRf+>

Parameter: Digits |MIN|MAX

Unit: mS|uS

Example: TRANSient:LTIME 500ms

Query Syntax: TRANSient:LTIME?

Parameter: None |MIN|MAX

Example: TRAN:LTIME? TRAN:LTIME? MIN TRAN:LTIME? MAX

Returned Parameter: <NR3>

Related Commands: TRAN:HTIM

### **TRANSient:RTIME**

Sets the time for transient rising edge.

Command Syntax: TRANSient:RTIME <NRf+>

Parameter: Digits |MIN|MAX

Unit: mS|uS

Example: TRANSient:RTIME 500ms

Query Syntax: TRANSient:RTIME?

Parameter: None |MIN|MAX

Example: TRAN:RTIME? TRAN:RTIME? MIN TRAN:RTIME? MAX

Returned Parameter: <NR3>

Related Commands: TRAN:FTIM

### **TRANSient:FTIME**

Sets the time for transient falling edge.

Command Syntax: TRANSient:FTIME <NRf+>

Parameter: Digits |MIN|MAX

Unit: mS|uS

Example: TRANSient:FTIME 200ms

Query Syntax: TRANSient:FTIME?

Parameter: None |MIN|MAX

Example: TRAN:FTIME? TRAN:FTIME? MIN TRAN:FTIME? MAX

Returned Parameter: <NR3>

Related Commands: TRAN:RTIM

## 2.2.8 LIST Subsystem

### [SOURCE:]LIST:CLEAr

Initializes the list data of the present list.

Command Syntax: LIST:CLEAr

Parameter: None

Example: LIST:CLEAr

### [SOURCE:]LIST[:STATe]

Enables/Disables the list operation.

Command Syntax: [SOURCE:]LIST[:STATe] <bool>

Parameter: On|Off

Example: LIST on LIST off

Query Syntax: [SOURCE:]LIST[:STATe]?

Returned Parameter: <NR1> Value:0 for OFF, 1 for ON

Related Commands: LIST:NUMB, LIST:CONT

### [SOURCE:]LIST:MEMO

Sets the Memo of the list.

Command Syntax: [SOURCE:]LIST: MEMO "<aard>"

Parameter: "0x20-0x7f"

Example: LIST: MEMO "GMC"

Query Syntax: [SOURCE:]LIST:MEMO?

Returned Parameter: <aard>

Related Commands: LIST:NUMB, LIST:CONT

### [SOURCE:]LIST:NUMBer

Sets the list number for the list operation.

Command Syntax: LIST:NUMBer <NR1>

Parameter: 0~6

Example: LIST:NUMB 2

Query Syntax: [SOURCE:]LIST:NUMBer?

Returned Parameter: <NR1> 0~6

Related Commands: LIST:ADD, LIST:CONT

### [SOURCE:]LIST[:STEP]:ADD

Adds a new step. The values for each new step include operating mode, load value, operating time.

Command Syntax: [SOURCE:]LIST[:STEP]:ADD <aard>,<NRf>,<NRf>

Parameter: CCL|CCH|CRL|CRM |CRH|CV, Digits |MIN|MAX, Digits |MIN|MAX

Unit: None, Current Unit| Voltage Unit| Resistance Unit, s

Example: LIST:ADD cch,1a,1s

Related Commands: LIST:DEL, LIST:INS

### [SOURCE:]LIST[:STEP]:INSert

Inserts a new step at specified step.

Command Syntax: [SOURCE:]LIST[:STEP]:INSert <NR1>,<aard>,<NRf>,<NRf>

Parameter: 1~50,CCL|CCH|CRL|CRM |CRH|CV, Digits |MIN|MAX, Digits |MIN|MAX

Unit: None, None, Current Unit| Voltage Unit| Resistance Unit, s

Example: LIST:INSert 2,cch,2a,1s

Related Commands: LIST:DEL, LIST:ADD

### [SOURCE:]LIST[:STEP]:EDIT

Edits one specified step.

Command Syntax: [SOURCE:]LIST[:STEP]:EDIT <NR1>,<aard>,<NRf>,<NRf>

Parameter: 1~50,CCL|CCH|CRL|CRM |CRH|CV, Digits|MIN|MAX, Digits|MIN|MAX

Unit: None, None, Current Unit| Voltage Unit| Resistance Unit, s

Example: LIST:EDIT 2,cch,3a,1s

### [SOURCE:]LIST[:STEP]:DELeTe

Deletes one specified step.

Command Syntax: [SOURCE:]LIST[:STEP]: DELeTe <NR1>

Parameter: 1~50

Unit: None

Example: LIST:DELeTe 2

Related Commands: LIST:ADD, LIST:INS

### [SOURCE:]LIST:CLEAr

Deletes all steps for list operation.

Command Syntax: [SOURCE:]LIST:CLEAr

Parameter: None

Example: LIST:SAVE

Related Commands: LIST:CLE

### [SOURCE:]LIST:COUNT

Sets the cycle times (Count) for list operation.

Command Syntax: [SOURCE:]LIST:COUNT <NR1>

Parameter: 0~65535

Unit: None

Example: LIST:COUNT 10

Query Syntax: LIST:COUNT?

Returned Parameter: <NR1>

Related Commands: LIST:NUMB, LIST:EDIT

### [SOURCE:]LIST:CHAIn

Sets the chained list for the present list.

Command Syntax: [SOURCE:]LIST:CHAIn <NR1>

Parameter: 0~6|OFF

Example: LIST:CHA 2

Query Syntax: LIST:CHA?

Returned Parameter: <NR1>

Related Commands: LIST:NUMB, LIST:MEMO

### [SOURCE:]LIST:SAVE

Saves the list date according to the list number.

Command Syntax: [SOURCE:]LIST:SAVE

Parameter: None

Example: LIST:SAVE

Related Commands: LIST:CLE

## 2.2.9 MODE

MODE

Sets the basic test mode for the load.

CC Mode CCL CCH

CV Mode CV

CP Mode CPC CPV

CR Mode CRL CRM CRH

Command Syntax: MODE <AARD>

Parameter: CCL|CCH|CRL| CRM|CRH|CV|CPC|CPV

Example: MODE CCL

Query Syntax: [SOURCE:]MODE?

Returned Parameter: <AARD> CCL|CCH|CRL| CRM|CRH|CV|CPC|CPV

## 2.2.10 MEASure

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

Queries the input voltage of the load.

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

Parameter: None

Query Syntax: MEAS:VOLT?

Returned Parameter: <NR3>

Related Commands: MEAS:CURR?, MEAS:RES?, MEAS:POW?

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

Queries the input current of the load.

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

Parameter: None

Query Syntax: MEAS:CURR?

Returned Parameter: <NR3>

Related Commands: MEAS:VOLT?, MEAS:RES?, MEAS:POW?

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

Queries the input power of the load.

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

Parameter: None

Query Syntax: MEAS:POW?

Returned Parameter: <NR3>

Related Commands: MEAS:CURR?, MEAS:VOLT?, MEAS:RES?

### MEASure[:SCALar]:RESistance[:DC]?

Queries the input resistance of the load.

Command Syntax: MEASure[:SCALar]:RESistance[:DC]?

Parameter: None

Query Syntax: MEAS:RES?

Returned Parameter: <NR3>

Related Commands: MEAS:CURR?, MEAS:VOLT?, MEAS:RES?

## 2.2.11 STATus Subsystem

### STATus:QUEStionable[:EVENT]?

This query reads the Questionable Status Event register, and returns a decimal value, which equals to the total weighted value of all binary bits. Reading this register resets it to zero. See Status Reporting for more details concerning the Questionable Status Event register.

Query Syntax: STATus:QUEStionable[:EVENT]?

Parameter: None

Returned Parameter: <NR1>

### STATus:QUEStionable:CONDition?

This query reads the Questionable Status Condition register. Each 0 to 1 transition of the condition of the Questionable Condition register will set the corresponding bit of the Questionable Status Event register. See Status Reporting for more details concerning the Questionable Status Condition register.

Query Syntax: STATus:QUEStionable:CONDition?

Parameter: None

Returned Parameter: <NR1>

### STATus:QUEStionable:ENABLE

Sets the bits of the Questionable Status Enable register. The programming parameters determine whether or not a 0 to 1 transition in the Questionable Event register will set the QUES bit of the Status Byte register. See Status Reporting for more details concerning the Questionable Status Enable register.

Command Syntax: STATus:QUEStionable:ENABLE <NRf>

Parameter: 0?65535

Power-on Value: refers to \*PSC command

Example: STATus:QUEStionable:ENABLE 64

Query Syntax: STATus:QUEStionable:ENABLE?

Returned Parameter: <NR1>

Related Commands: \*PSC

### STATus:OPERation[:EVENT]?

This query reads the Operation Event register. Reading this register resets it to zero. See Status Reporting for more details concerning the Operation Event register.

Query Syntax: STATus:OPERation[:EVENT]?

Parameter: None

Returned Parameter: <NR1>

### STATus:OPERation:CONDition?

This query reads the Operation Status Condition register. Each 0 to 1 transition of the condition of the Operation Condition register will set the corresponding bit of the Operation Event register. See Status Reporting for more details concerning the Operation Status Condition register.

Query Syntax: STATus:OPERation:CONDition?

Parameter: None

Returned Parameter: <NR1>

### STATus:OPERation:ENABLE

Sets the bits of the Operation Status Enable register. The programming parameters determine whether or not a 0 to 1 transition in the Operation Event register will set the OPER bit of the Status Byte register. See Status Reporting for more details concerning the Operation Status Enable register.

Command Syntax: STATus:OPERation:ENABLE <NRf>

Parameter: 0?255

Power-on Value: refers to \*PSC command

Example: STATus:OPERation:ENABLE 128

Query Syntax: STATus:OPERation:ENABLE?

Returned Parameter: <NR1>

Related Commands: \*PSC

## 2.2.12 SYSTem

### SYSTem:ERRor[:NEXT]?

Queries the error code and error message of the load. When the "ERR" annunciator on the front panel is turned on, one or more errors have occurred to the load. At most 20 error records can be saved in the load. It will be not saved if the error records exceed 20.

- 1 The error queue operates in a FIFO (first-in, first-out) mode to save and read the error. The first returned error is the first saved error. When all errors have been read, the "ERR" annunciator will be turned off automatically.
- 2 If the error records exceed 20, the last error in the error queue will be replaced by 350, "Too many errors". The load will not save any additional error message, unless you clear or read errors from the queue.
- 3 If there is no error occurs to the executing command, "No error" will be returned.

\*CLS can clear the error queue, while \*RST can not. All error records will be lost after the load is turned off.

Command Syntax: SYST:ERR[:NEXT]?

Parameter: None

Returned Parameter: <NR1>?<AARD>

### SYSTem:VERSion?

Queries the software version number, e.g. V1.00.

Command Syntax: SYST:VERS?

Parameter: None

Returned Parameter: <AARD>?<NR2>

### SYSTem:REMOte

Sets the load in Remote status. An error message will be generated when the load is communicated via GPIB port.

Command Syntax: SYSTem:REMOte

Parameter: None

Returned Parameter: None

Related Commands: SYSTem:LOCal

### SYSTem:LOCal

Sets the load in Local status. An error message will be generated when the load is communicated via GPIB port.

Command Syntax: SYSTem:LOCAl

Parameter: None

Returned Parameter: None

Related Commands: SYSTem:REMOte

## 2.2.13 TRIGger Subsystem

### TRIGger[:IMMediate]

This command generates a trigger signal to the load, regardless of which trigger source is currently in effect. It has the same function as the \*TRG command.

Command Syntax: TRIGger[:IMMediate]

Parameter: None

Related Commands: TRIG:SOUR

### TRIGger:SOURce

Selects the load trigger source. There are three trigger sources: BUS, EXTERNAL, HOLD.

BUS: Accepts a GPIB <GET> signal or \*TRG command as the trigger source.

EXTERNAL: Selects the External trigger input terminal or key on the front panel as the trigger source. The input signal at the External trigger input terminal is TTL; the falling edge is triggered.

HOLD: Only the TRIGger:IMMediate command causes a trigger in this mode. All others, including \*TRG, are ignored.

Command Syntax: TRIGger:SOURce <aard>

Parameter: BUS| EXTERNAL| HOLD

Example: TRIG:SOUR EXT

Query Syntax: TRIGger:SOURce?

Returned Parameter: <aard>

### TRIGger:FUNCTion

Sets Transient (Tran) or List function for the trigger object.

Command Syntax: TRIGger:FUNCTion <aard>

Parameter: Tran| List

Query Syntax: TRIGger:FUNCTion?

Example: TRIG:FUNC LIST

Returned Parameter: <aard>Tran| List

### INITiate[:IMMediate]

Activates the trigger input terminal once. After a trigger signal has been received it is automatically deactivated again

Command Syntax: INITiate [:IMMediate]

Parameter: None

Example: INIT

Related Commands: \*TRG

### INITiate:CONTinuous

Activates the trigger input terminal permanently.

Command Sytax: INITiate:CONTinuous 1

Parameter: ON(1) OFF(0)

Example: INIT:CONT1

Returned Parameter: <NR1> Value: 0 for OFF, 1 for ON

Related Commands: \*TRG

## 2.2.14 ATTeRy Subsystem

### [SOURce:]BATTery[:STATe]

Enables/Disables the Battery Discharge operation.  
 Command Syntax: [SOURce:]BATTery[:STATe] <bool>  
 Parameter: ON|OFF  
 Example: BATT on;  
 Query Syntax: [SOURce:]BATTery[:STATe]?  
 Example: BATT?  
 Returned Parameter: <NR1> Value:0(OFF) 1(ON)  
 Related Commands: INP ON

### [SOURce:]BATTery:TERMinate:VOLTage

Sets termination voltage value for the battery discharge operation. The battery voltage can not be less than the termination voltage value, or the battery discharge test will be stopped automatically.  
 Command Syntax: [SOURce:]BATTery:TERMinal:VOLTage <NRf+>  
 Parameter: Digits |MIN|MAX  
 Unit: V|mV  
 Example: BATT:TERM:VOLT 2V  
 Parameter: None |MIN|MAX  
 Query Syntax: [SOURce:]BATTery:TERMinal:VOLTage:?  
 Example: BATT:TERM:VOLT? BATT:TERM:VOLT? MAX;  
 BATT:TERM:VOLT? MIN  
 Returned Parameter: <NR3>  
 Related Commands: VOLT?

### [SOURce:]BATTery:DISCharge:CURREnt

Sets the battery discharge current.  
 Command Syntax: [SOURce:]BATTery:DISCharge:CURREnt <NRf+>  
 Parameter: Digits |MIN|MAX  
 Unit: A|mA  
 Example: BATT:CURR 1A  
 Query Syntax: [SOURce:]BATTery[:DISCharge]:CURREnt?  
 Parameter: None |MIN|MAX  
 Returned Parameter: <NR3>  
 Related Commands: BATT:VOLT

### [SOURce:]BATTery:CAPAcity:CLEar

Queries the discharged capacity of the battery.  
 Command Syntax: BATTery:CAPAcity:CLEar  
 Parameter: None  
 Example: BATT:CAPA:CLE  
 Returned Parameter: <NR3>

### [SOURce:]BATTery[:DISCharge]:TIME?

Queries the discharged time of the battery.  
 Command Syntax: [SOURce:]BATTery[:DISCharge]:TIME?  
 Example: BATT:TIME?  
 Returned Parameter: <NR1>:<NR1>:<NR1>(Hour: Minute: Second)  
 Related Commands: BATT:VOLT

## 3 Error Messages

Table 3-1 lists the error numbers and associated error messages that apply to the electronic load.

Table 3-1. Summary of Error Messages

Error Number	Error Description
-103	<b>Invalid separator</b> There is invalid separator existed in command string. You may use the comma to replace the colon, semicolon, or space; or you may use the space to replace the colon.
-104	<b>Data type error</b> You may specify a digit where the character string is required, or you may specify a character string where a digit is required.
-108	Missing parameter, or Parameter not allowed. Received few parameters. One or more parameters are missing; Received too many parameters. You may have input extra parameters, or added the parameter into the command where no parameter is allowed.
-113	<b>Undefined header</b> You may have misspelled the command, or the command is invalid. If you use the command of short form, only four letters can be included at most.
-123	<b>Exponent too large</b>
-131	<b>Invalid suffix</b> You may misspell a wrong suffix.
-170	<b>Expression error</b> You may start an expression with single quotation mark, and end it with a double quotation mark.
-210	<b>Command not allowed with GPIB</b>
-221	<b>LIST out of range</b>
-222	<b>Data out of range</b>
-223	<b>Too much data</b> Received the character string, but it can not be executed because the string length exceeds the allowed character numbers.
-330	<b>Self-test failed</b> One or more errors are detected when it is self-testing.
-350	<b>Too many errors</b> The existing errors exceed 20, and the error queue becomes full. The error queue cannot save any new error unless the existing errors have been deleted. Turning off the power or executing *CLS command will clear the error queue.
-410	<b>Query INTERRUPTED</b> Receive the command to send data to the output buffer, but the data sent by the previous command is contained in the output buffer (the previous data is not covered). Turning off the power or executing *RST command will clear the output buffer.
-440	<b>Query UNTERMINATED</b> Addressed to talk (sending data via the interface), but the command of sending data to the output buffer is not received.

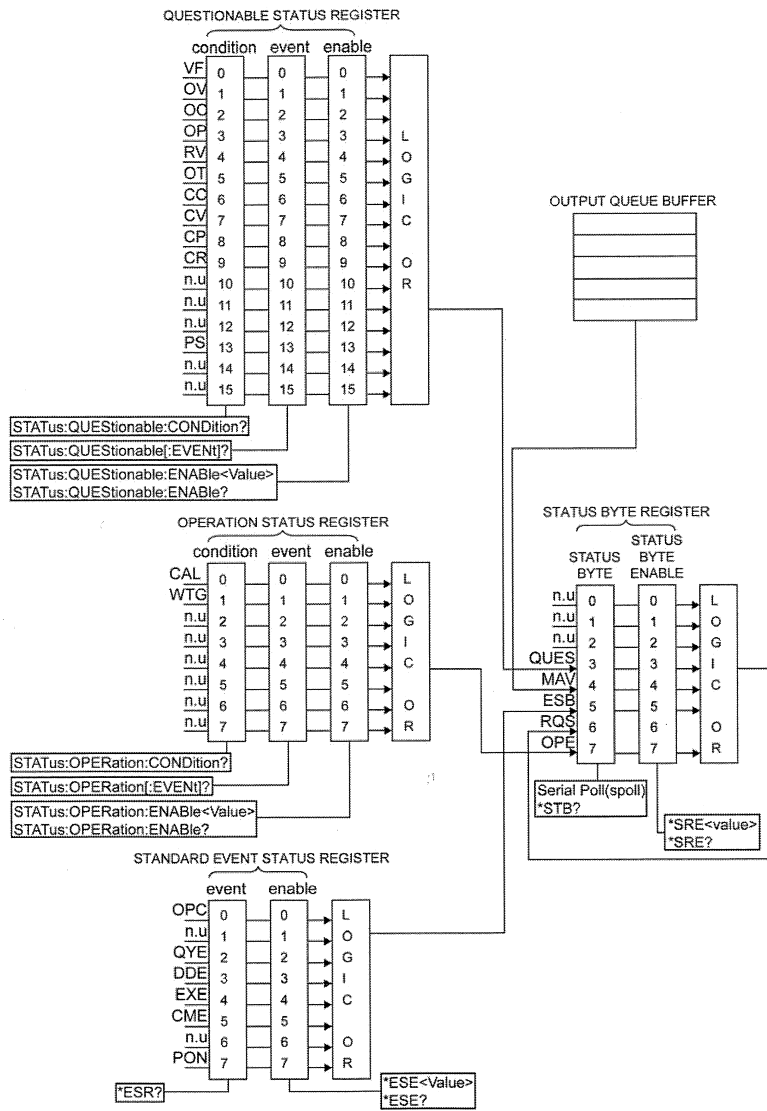
## 4 Status Register Reporting

### SCPI Status Register

The electronic load has four status register to record its status. They are the Status Byte register, the Standard Event register, the Questionable Status register, and the Operation Status register. The Status Byte register summarizes all of the status events from all status registers. The list below is the description for all status registers.

BIT	Signal	Meaning
		<b>Operation Status register</b>
0	CAL	A calibration calculation is in progress.
1	WTG	Waiting for trigger.
		<b>Questionable Status register</b>
0	VF	<b>Voltage Error</b> It is caused by the reversed voltage or the over-voltage; VF remains set till the "INP:PROT:CLE" is received.
1	OV	<b>Over-voltage</b> When the Over-voltage occurs, the load is turned off, and OV/OV is set at the same time. OV/OV remain set till the over-voltage condition is removed and the "INP:PROT:CLE" is received.
2	OC	<b>Over-current</b> When the current exceeds the protection current limit, OC is set, and remains set till the over-current condition is removed. However, if OC remain set for the specified delay period, PS will be set, and the load is turned off. In this case, PS/OC remain set till the over-current condition is removed, and the "INP:PROT:CLE" is received.
3	OP	<b>Over-power</b> When over-power occurs, the load is turned off, and OP/PS are set. OP/PS remain set till the over-power condition is removed, and the "INP:PROT:CLE" is received.
4	RV	<b>Reverse Voltage on input</b> When reverse voltage occurs, RV/VF are set. RV is reset after the reverse voltage is removed, but VF remains set till the "INP:PROT:CLE" is received.
5	OT	<b>Over-temperature</b> When over-power occurs, the load is turned off, and OP/PS are set. OP/PS remain set till the temperature is reduced by the fan and the "INP:PROT:CLE" is received.
6	CC	<b>Constant Current mode</b>
7	CV	<b>Constant Voltage mode</b>
8	CP	<b>Constant Power mode</b>
9	CR	<b>Constant Resistance mode</b>
13	PS	<b>Protection Shutdown</b> The electronic load is turned off due to over-current, over-power or over-temperature; the PS is set and remain set till the "INP:PROT:CLE" is received.
		<b>Standard Event register</b>
0	OPC	<b>Operation Complete</b> The load has completed all pending operations. Programming *OPC causes this bit to be set when the load completes all pending operations.
2	QYE	<b>Query Error</b> The output queue was read when no data was present. Errors in the range of -499 to -400 can set QYE.
3	DDE	<b>Device Dependent Error</b> Memory was lost. Errors in the range of -399 to -300 can set DDE.
4	EXE	<b>Execution Error</b> A command parameter was outside the legal range or inconsistent with the electronic load's operation, or the command could not be executed due to some operating condition. Errors in the range of -299 to -200 can set EXE.
5	CME	<b>Command Error</b> A syntax or semantic error has occurred. Errors in the range of -199 to -100 can set CME.
7	PON	<b>Power On</b> This bit is always set when the load is turned on.
		<b>Status Byte register</b>
3	QUES	<b>Questionable</b> If an enabled questionable status register is affected, the QUES is set.
4	MAV	<b>Message Available</b> If the Output Queue contains data, the MAV is set.
5	ESB	<b>Event Status Bit</b> If a Standard Event Enable register is affected, the ESB is set.
6	MSS RQS	During a serial poll, RQS (Request Service) is returned and cleared. For an *STB? query, MSS (Master Summary Status) is returned without being cleared.
7	OPER	<b>Operation</b> If an enabled Operation Status register is affected, the OPER is set.

The following diagram shows the relationship among the load's status registers.



## 5 Product Support

If required please contact:

Gossen Metrawatt GmbH

**Product Support Hotline**

Phone: +49-911-8602-0

Fax: +49 911 8602-709

E-mail: [support@gossenmetrawatt.com](mailto:support@gossenmetrawatt.com)

---

© Gossen Metrawatt GmbH

Edited in Germany • Subject to change, errors excepted • PDF version available on the Internet

All trademarks, registered trademarks, logos, product names, and company names are the property of their respective owners.

 **GOSSEN METRAWATT**

Gossen Metrawatt GmbH  
Südwestpark 15  
90449 Nürnberg • Germany

Telefon +49 911 8602-111  
Telefax +49 911 8602-777  
E-Mail [info@gossenmetrawatt.com](mailto:info@gossenmetrawatt.com)  
[www.gossenmetrawatt.com](http://www.gossenmetrawatt.com)