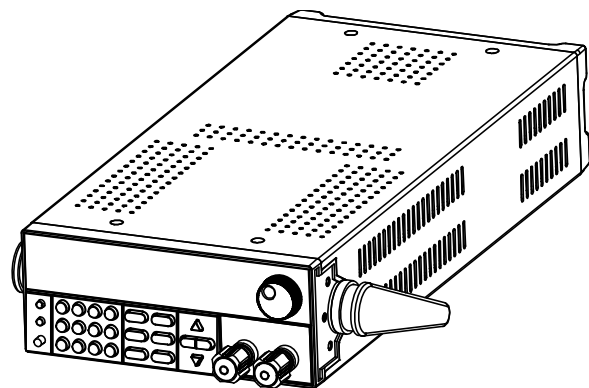


DC Programmable Electronic Load

IT8500G+ Series Programming Guide



Model: IT8511G+/IT8511AG+/IT8512G+
/IT8512BG+/IT8513G+/IT8513BG+
/IT8513CG+

Revision: 3.0

Notices

© Itech Electronic, Co., Ltd. 2023

No part of this manual may be reproduced in any form or by any means (including electronic storage and retrieval or translation into a foreign language) without prior permission and written consent from Itech Electronic, Co., Ltd. as governed by international copyright laws.

Manual Part Number

IT8500G+-402225

Revision

Second Edition: Apr. 12,
2023

Itech Electronic, Co., Ltd.

Trademarks

Pentium is U.S. registered trademarks of Intel Corporation.

Microsoft, Visual Studio, Windows and MS Windows are registered trademarks of Microsoft Corporation in the United States and/or other countries and regions.

Warranty

The materials contained in this document are provided “as is”, and is subject to change, without prior notice, in future editions. Further, to the maximum extent permitted by applicable laws, ITECH disclaims all warranties, either express or implied, with regard to this manual and any information contained herein, including but not limited to the implied warranties of merchantability and fitness for a particular purpose. ITECH shall not be held liable for errors or for incidental or indirect damages in connection with the furnishing, use or application of this document or of any information contained herein. Should ITECH and the user enter into a separate written agreement with warranty terms covering the materials in this document that conflict with these terms, the warranty terms in the separate agreement shall prevail.

Technology Licenses

The hardware and/or software described herein are furnished under a license and may be used or copied only in accordance with the terms of such license.

Restricted Rights Legend

Restricted permissions of the U.S. government. Permissions for software and technical data which are authorized to the U.S. Government only include those for custom provision to end users. ITECH follows FAR 12.211 (technical data), 12.212 (computer software). DFARS 252.227-7015 (technical data-- commercial products) for national defense and DFARS 227.7202-3 (permissions for commercial computer software or computer software documents) while providing the customized business licenses of software and technical data.

Safety Notices

CAUTION

A CAUTION sign denotes a hazard. It calls attention to an operating procedure or practice that, if not correctly performed or adhered to, could result in damage to the product or loss of important data. Do not proceed beyond a CAUTION sign until the indicated conditions are fully understood and met.

WARNING

A WARNING sign denotes a hazard. It calls attention to an operating procedure or practice that, if not correctly performed or adhered to, could result in personal injury or death. Do not proceed beyond a WARNING sign until the indicated conditions are fully understood and met.



NOTE

A NOTE sign denotes important hint. It calls attention to tips or supplementary information that is essential for users to refer to.

Quality Certification and Assurance

We certify that series IT8500G+ electronic load meets all the published specifications at time of shipment from the factory.

Warranty

ITECH warrants that the product will be free from defects in material and workmanship under normal use for a period of one (1) year from the date of delivery (except those described in the Limitation of Warranty below).

For warranty service or repair, the product must be returned to a service center designated by ITECH.















- The product returned to ITECH for warranty service must be shipped PREPAID. And ITECH will pay for return of the product to customer.
- If the product is returned to ITECH for warranty service from overseas, all the freights, duties and other taxes shall be on the account of customer.


Limitation of Warranty

This Warranty will be rendered invalid in case of the following:

- Damage caused by circuit installed by customer or using customer own products or accessories;
- Modified or repaired by customer without authorization;
- Damage caused by circuit installed by customer or not operating our products under designated environment;
- The product model or serial number is altered, deleted, removed or made illegible by customer;
- Damaged as a result of accidents, including but not limited to lightning, moisture, fire, improper use or negligence.

Safety Symbols

	Direct current		ON (power on)
	Alternating current		OFF (power off)
	Both direct and alternating current		Power-on state
	Protective conductor terminal		Power-off state
	Earth (ground) terminal		Reference terminal
	Caution, risk of electric shock		Positive terminal
	Warning, risk of danger (refer to this manual for specific Warning or Caution information)		Negative terminal

	Frame or chassis terminal	-	-
---	---------------------------	---	---

Safety Precautions

The following safety precautions must be observed during all phases of operation of this instrument. Failure to comply with these precautions or specific warnings elsewhere in this manual will constitute a default under safety standards of design, manufacture and intended use of the instrument. ITECH assumes no liability for the customer's failure to comply with these precautions.

WARNING

- Series IT8500G+ electronic load supports 110V/220VAC input and need to switch the input voltage before operation.
- Do not use the instrument if it is damaged. Before operation, check the casing to see whether it cracks. Do not operate the instrument in the presence of inflammable gasses, vapors or dusts.
- The electronic load is provided with a power line during delivery and should be connected to a socket with a protective earth terminal. Before operation, be sure that the instrument is well grounded.
- Make sure to use the power cord supplied by ITECH.
- Check all marks on the instrument before connecting the instrument to power supply.
- Use electric wires of appropriate load. All loading wires should be capable of bearing maximum short-circuit current of electronic load without overheating. If there are multiple electronic loads, each pair of the power cord must be capable of bearing the full-loaded rated short-circuit output current
- Ensure the voltage fluctuation of mains supply is less than 10% of the working voltage range in order to reduce risks of fire and electric shock.
- Do not install alternative parts on the instrument or perform any unauthorized modification.
- Do not use the instrument if the detachable cover is removed or loosen.
- To prevent the possibility of accidental injuries, be sure to use the power adapter supplied by the manufacturer only.
- We do not accept responsibility for any direct or indirect financial damage or loss of profit that might occur when using the instrument.
- This instrument is used for industrial purposes, do not apply this product to IT power supply system.
- Never use the instrument with a life-support system or any other equipment subject to safety requirements.

CAUTION

- Failure to use the instrument as directed by the manufacturer may render its protective features void.
- Always clean the casing with a dry cloth. Do not clean the internals.
- Make sure the vent hole is always unblocked.

Environmental Conditions

The instrument is designed for indoor use and an area with low condensation. The table below shows the general environmental requirements for the instrument. The speed of fan will change intelligently by the temperature of radiator. When the temperature is up to 40°C, the fan will be on and adjust intelligently when temperature changes.




Environmental Conditions	Requirements
Operating temperature	0°C to 40°C
Operating humidity	20%-80% (non-condensation)
Storage temperature	-20°C to 70 °C
Altitude	Operating up to 2,000 meters
Pollution degree	Pollution degree 2
Installation category	II



Note

To make accurate measurements, allow the instrument to warm up for 30 min before operation.

Regulatory Markings

	The CE mark indicates that the product complies with all the relevant European legal directives. The specific year (if any) affixed refers to the year when the design was approved.
	The instrument complies with the WEEE Directive (2002/96/EC) marking requirement. This affixed product label indicates that you must not discard the electrical/electronic product in domestic household waste.
	This symbol indicates the time period during which no hazardous or toxic substances are expected to leak or deteriorate during normal use. The expected service life of the product is 10 years. The product can be used safely during the 10-year Environment Friendly Use Period (EFUP). Upon expiration of the EFUP, the product must be immediately recycled.

Waste Electrical and Electronic Equipment (WEEE) Directive



2002/96/EC Waste Electrical and Electronic Equipment (WEEE) Directive

This product complies with the WEEE Directive (2002/96/EC) marking requirement. This affix product label indicates that you must not discard the electrical/electronic product in domestic household waste.

Product Category

With reference to the equipment classifications described in the Annex I of the WEEE Directive, this instrument is classified as a “Monitoring and Control Instrument”.

To return this unwanted instrument, contact your nearest ITECH office.

Compliance Information

Complies with the essential requirements of the following applicable European Directives, and carries the CE marking accordingly:

- Electromagnetic Compatibility (EMC) Directive 2014/30/EU
- Low-Voltage Directive (Safety) 2014/35/EU

Conforms with the following product standards:

EMC Standard

IEC 61326-1:2012/ EN 61326-1:2013 ¹²³

Reference Standards

CISPR 11:2009+A1:2010/ EN 55011:2009+A1:2010 (Group 1, Class A)

IEC 61000-4-2:2008/ EN 61000-4-2:2009

IEC 61000-4-3:2006+A1:2007+A2:2010/ EN 61000-4-3:2006+A1:2008+A2:2010

IEC 61000-4-4:2004+A1:2010/ EN 61000-4-4:2004+A1:2010

IEC 61000-4-5:2005/ EN 61000-4-5:2006

IEC 61000-4-6:2008/ EN 61000-4-6:2009

IEC 61000-4-11:2004/ EN 61000-4-11:2004

1. The product is intended for use in non-residential/non-domestic environments. Use of the product in residential/domestic environments may cause electromagnetic interference.
2. Connection of the instrument to a test object may produce radiations beyond the specified limit.
3. Use high-performance shielded interface cable to ensure conformity with the EMC standards listed above.

Safety Standard

IEC 61010-1:2010/ EN 61010-1:2010

Content

Quality Certification and Assurance	i
Warranty	i
Limitation of Warranty.....	i
Safety Symbols.....	i
Safety Precautions	ii
Environmental Conditions.....	iii
Regulatory Markings.....	iii
Waste Electrical and Electronic Equipment (WEEE) Directive.....	iv
Compliance Information.....	v
Chapter 1 Remote Control.....	1
1.1 Overview	1
1.2 SCPI Command Introduction.....	1
1.3 Command Type of SCPI.....	1
1.4 Message Type of SCPI.....	3
1.5 Response Data Type	5
1.6 Command Format.....	6
1.7 Data Type	8
1.8 Remote Interface Communication	9
Chapter 2 SCPI Status Register	10
Chapter 3 System Commands	13
SYSTem:POSetup	13
SYSTem:POSetup?	13
SYSTem:BEEPer:IMMEDIATE	13
SYSTem:MEMory	14
SYSTem:MEMory?	14
SYSTem:COMMunicate:SElect	14
SYSTem:COMMunicate:SElect?	15
SYSTem:RUNMode	15
SYSTem:RUNMode?	16
SYSTem:PRESet.....	16
SYSTem:DEFault.....	16
SYSTem:CLEar	17
SYSTem:ERRor[:NEXT]?	17
SYSTem:VERSion?	17
SYSTem:SENSe[:STATe] <bool>	18
SYSTem:SENSe[:STATe]?.....	18
SYSTem:LOCal	19
SYSTem:REMote.....	19
SYSTem:RWLock	19
Chapter 4 Multi-channel commands	21
SYSTem:CHANnel:NUMBer.....	21
SYSTem:CHANnel:NUMBer?.....	21
LINK:MODE	21
LINK:MODE?	22
CHANnel.....	22
CHANnel?	23
Chapter 5 Status Commands	24
STATus:QUEStionable[:EVENT]?.....	24
STATus:QUEStionable:CONDition?	24
STATus:QUEStionable:ENABle <NR1>	25
STATus:QUEStionable:ENABle?	25
STATus:OPERation[:EVENT]?	25
STATus:OPERation:CONDition?	26
STATus:OPERation:ENABle <NR1>	26

STATus:OPERation:ENABLE?	27
Chapter 6 Input Settings.....	28
[SOURce:]INPut <bool>	28
[SOURce:]INPut?	28
[SOURce:]INPut:SHORt <bool>	28
[SOURce:]INPut:SHORt?	29
[SOURce:]INPut:TIMer[:STATe] <bool>	29
[SOURce:]INPut:TIMer[:STATe]?	30
[SOURce:]INPut:TIMer:DELay <NRf+>	30
[SOURce:]INPut:TIMer:DELay?	30
[SOURce:]FUNction:MODE <CPD>	30
[SOURce:]FUNction:MODE?	31
[SOURce:]FUNction <function>	31
[SOURce:]MODE <function>	31
[SOURce:]FUNction?	32
[SOURce:]MODE?	32
[SOURce:]TRANsient[:STATe] <BOOL>	32
[SOURce:]TRANsient[:STATe]?	33
OUTPut:INHibit:MODE <CRD>	33
OUTPut:INHibit:MODE?	33
INPut:SDS:SURGe:SUPPress <BOOL>	34
INPut:SDS:SURGe:SUPPress?	34
[SOURce:]CURRent:RANGe <NRf+>	34
[SOURce:]CURRent:RANGe?	35
[SOURce:]VOLTagE:RANGe <NRf+>	35
[SOURce:]VOLTagE:RANGe?	36
[SOURce:]VOLTagE:RANGe:AUTO[:STATe]	36
[SOURce:]CURRent:SLEW[:BOTH] <NRf+>	37
[SOURce:]CURRent:SLEW[:BOTH]?	37
[SOURce:]CURRent:SLEW:RISE <NRf+>	37
[SOURce:]CURRent:SLEW:POSitive <NRf+>	37
[SOURce:]CURRent:SLEW:RISE?	38
[SOURce:]CURRent:SLEW:POSitive?	38
[SOURce:]CURRent:SLEW:FALL <NRf+>	38
[SOURce:]CURRent:SLEW:NEGative <NRf+>	38
[SOURce:]CURRent:SLEW:FALL?	39
[SOURce:]CURRent:SLEW:NEGative?	39
[SOURce:]CURRent[:OVER]:PROTection:STATe <BOOL>	39
[SOURce:]CURRent[:OVER]:PROTection:STATe?	40
[SOURce:]CURRent:PROTection[:LEVel] <NRf+>	40
[SOURce:]CURRent:PROTection[:LEVel]?	40
[SOURce:]CURRent[:OVER]:PROTection:DELay <NRf+>	41
[SOURce:]CURRent[:OVER]:PROTection:DELay?	41
[SOURce:]POWer:PROTection[:LEVel] <NRf+>	41
[SOURce:]POWer:PROTection[:LEVel]?	42
[SOURce:]POWer:PROTection:DELay <NRf+>	42
[SOURce:]POWer:PROTection:DELay?	43
[SOURce:]POWer:CONFig[:LEVel] <NRf+>	43
[SOURce:]POWer:CONFig[:LEVel]?	43
[SOURce:]VOLTagE:[LEVel:]ON <NRf+>	44
[SOURce:]VOLTagE:[LEVel:]ON?	44
[SOURce:]CURRent:TRANsient:MODE <CRD>	44
[SOURce:]DYNamic:MODE <CRD>	44
[SOURce:]CURRent:TRANsient:MODE?	45
[SOURce:]DYNamic:MODE?	45
[SOURce:]VOLTagE:[LEVel:]OFF <NRf+>	45
[SOURce:]VOLTagE:[LEVel:]OFF?	46
[SOURce:]VOLTagE:LATCh[:STATe] <BOOL>	46
[SOURce:]VOLTagE:LATCh[:STATe]?	46
[SOURce:]VOLTagE:ILIMIT <NRf+>	47
[SOURce:]VOLTagE:ILIMIT?	47

[SOURCE:]VOLTage:SLOPe[:RANGe] <CRD>	47
[SOURCE:]VOLTage:SLOPe[:RANGe]?	48
[SOURCE:]VOLTage:TRANSient:MODE <CRD>	48
[SOURCE:]VOLTage:TRANSient:MODE?	49
[SOURCE:]VOLTage:TRANSient:ALEVel <NRf+>	49
[SOURCE:]VOLTage:TRANSient:ALEVel?	49
[SOURCE:]VOLTage:TRANSient:BLEVel <NRf+>	49
[SOURCE:]VOLTage:TRANSient:BLEVel?	50
[SOURCE:]VOLTage:TRANSient:AWIDth <NRf+>	50
[SOURCE:]VOLTage:TRANSient:AWIDth?	51
[SOURCE:]VOLTage:TRANSient:BWIDth <NRf+>	51
[SOURCE:]VOLTage:TRANSient:BWIDth?	51
[SOURCE:]CURRent[:LEVel][:IMMediate][:AMPLitude] <NRf+>	52
[SOURCE:]CURRent[:LEVel][:IMMediate][:AMPLitude]?	52
[SOURCE:]VOLTage[:LEVel][:IMMediate][:AMPLitude] <NRf+>	52
[SOURCE:]VOLTage[:LEVel][:IMMediate][:AMPLitude]?	53
[SOURCE:]POWer[:LEVel][:IMMediate][:AMPLitude] <NRf+>	53
[SOURCE:]POWer[:LEVel][:IMMediate][:AMPLitude]?	54
[SOURCE:]POWer:ILIMIT <NRf+>	54
[SOURCE:]POWer:ILIMIT?	55
[SOURCE:]POWer:TRANSient:MODE <CRD>	55
[SOURCE:]POWer:TRANSient:MODE?	55
[SOURCE:]POWer:TRANSient:ALEVel <NRf+>	56
[SOURCE:]POWer:TRANSient:ALEVel?	56
[SOURCE:]POWer:TRANSient:BLEVel <NRf+>	56
[SOURCE:]POWer:TRANSient:BLEVel?	57
[SOURCE:]POWer:TRANSient:AWIDth <NRf+>	57
[SOURCE:]POWer:TRANSient:AWIDth?	57
[SOURCE:]POWer:TRANSient:BWIDth <NRf+>	57
[SOURCE:]POWer:TRANSient:BWIDth?	58
[SOURCE:]RESistance[:LEVel][:IMMediate][:AMPLitude] <NRf+>	58
[SOURCE:]RESistance[:LEVel][:IMMediate][:AMPLitude]?	59
[SOURCE:]RESistance:ILIMIT <NRf+>	59
[SOURCE:]RESistance:ILIMIT?	59
[SOURCE:]RESistance:TRANSient:MODE <CRD>	60
[SOURCE:]RESistance:TRANSient:MODE?	60
[SOURCE:]RESistance:TRANSient:ALEVel <NRf+>	60
[SOURCE:]RESistance:TRANSient:ALEVel?	61
[SOURCE:]RESistance:TRANSient:BLEVel <NRf+>	61
[SOURCE:]RESistance:TRANSient:BLEVel?	61
[SOURCE:]RESistance:TRANSient:AWIDth <NRf+>	62
[SOURCE:]RESistance:TRANSient:AWIDth?	62
[SOURCE:]RESistance:LED[:STATe] <BOOL>	62
[SOURCE:]RESistance:LED[:STATe]?	63
[SOURCE:]RESistance:TRANSient:BWIDth <NRf+>	63
[SOURCE:]RESistance:TRANSient:BWIDth?	63
[SOURCE:]DYNamic:HIGH[:LEVel] <NRf+>	63
[SOURCE:]CURRent:TRANSient:ALEVel <NRf+>	63
[SOURCE:]DYNamic:HIGH[:LEVel]?	64
[SOURCE:]CURRent:TRANSient:ALEVel?	64
[SOURCE:]DYNamic:HIGH:DWELI <NRf+>	65
[SOURCE:]CURRent:TRANSient:AWIDth <NRf+>	65
[SOURCE:]DYNamic:HIGH:DWELI?	65
[SOURCE:]CURRent:TRANSient:AWIDth?	65
[SOURCE:]DYNamic:LOW[:LEVel] <NRf+>	66
[SOURCE:]CURRent:TRANSient:BLEVel <NRf+>	66
[SOURCE:]DYNamic:LOW[:LEVel]?	66
[SOURCE:]CURRent:TRANSient:BLEVel?	66
[SOURCE:]DYNamic:LOW:DWELI <NRf+>	67
[SOURCE:]CURRent:TRANSient:BWIDth <NRf+>	67
[SOURCE:]DYNamic:LOW:DWELI?	67
[SOURCE:]CURRent:TRANSient:BWIDth?	67

Chapter 7	Measure Commands	68
	MEASure[:SCALar]:VOLTage[:DC]?	68
	FETCh[:SCALar]:VOLTage[:DC]?	68
	MEASure[:SCALar]:VOLTage:MAXimum?	68
	FETCh[:SCALar]:VOLTage[:AMPLitude]:MAXimum[:INSTant]?	68
	MEASure[:SCALar]:VOLTage:MINimum?	69
	FETCh[:SCALar]:VOLTage[:AMPLitude]:MINimum[:INSTant]?	69
	MEASure[:SCALar]:CURRent[:DC]?	69
	FETCh[:SCALar]:CURRent[:DC]?	69
	MEASure[:SCALar]:CURRent:MAXimum?	69
	FETCh[:SCALar]:CURRent[:AMPLitude]:MAXimum[:INSTant]?	69
	MEASure[:SCALar]:CURRent:MINimum?	70
	FETCh[:SCALar]:CURRent[:AMPLitude]:MINimum[:INSTant]?	70
	MEASure[:SCALar]:POWer[:DC]?	70
	FETCh[:SCALar]:POWer[:DC]?	70
	MEASure[:SCALar]:RESistance[:DC]?	71
	FETCh[:SCALar]:RESistance[:DC]?	71
	MEASure:CAPacity?	71
	FETCh:CAPacity?	71
	MEASure:TIME?	72
	FETCh:TIME?	72
	MEASure:TEMPerature?	72
	FETCh:TEMPerature?	72
	MEASure:VOLTage:RIPple?	73
	MEASure[:SCALar]:VOLTage:PTPeak?	73
	MEASure:CURRent:RIPple?	73
	MEASure[:SCALar]:CURRent:PTPeak?	73
Chapter 8	TRIGger Commands	74
	TRIGger[:IMMediate]	74
	TRIGger:SOURce <CPD>	74
	TRIGger:SOURce?	74
Chapter 9	SENSe Commands	76
	SENSe:AVERage <NRf+>	76
	SENSe:AVERage?	76
	SENSe:TIME:VOLTage1 <NRf+>	76
	SENSe:TIME:VOLTage1?	76
	SENSe:TIME:VOLTage2 <NRf+>	77
	SENSe:TIME:VOLTage2?	77
	SENSe[:REMote][:STATe] <bool>	77
	SENSe[:REMote][:STATe]?	78
	SENSe:TIME:VOLTage:UP?	78
	SENSe:TIME:VOLTage:DOWN?	78
	SENSe:VOLTage:POSitive:PULSe?	78
	SENSe:VOLTage:NEGative:PULSe?	79
Chapter 10	List Commands	80
	[SOURce:]LIST:RANGe <NRf+>	80
	[SOURce:]LIST:RANGe?	80
	[SOURce:]LIST:COUNT < NRf+>	80
	[SOURce:]LIST:COUNT?	81
	[SOURce:]LIST:STEP < NRf+>	81
	[SOURce:]LIST:STEP?	81
	[SOURce:]LIST:SLEW[:BOTH] <NR1>,<NRf+>	81
	[SOURce:]LIST:SLEW[:BOTH]? <NR1>	82
	[SOURce:]LIST:LEVel <NR1>,<NRf+>	82
	[SOURce:]LIST:LEVel? <NR1>	82
	[SOURce:]LIST:WIDTh <NR1>,<NRf+>	83
	[SOURce:]LIST:WIDT?	83
	[SOURce:]LIST:SAV < NR1>	83
	[SOURce:]LIST:RCL < NR1>	84

[SOURce:]LIST:RUN:STEP?	84
[SOURce:]LIST:RUN:REPeat?	84
Chapter 11 QC Commands	85
QC2QC INIT	85
QC2QC:classA <CPD>	85
QC2QC:classA?	85
QC2QC:classB <CPD>	85
QC2QC:classB?	86
QC3QC INIT	86
QC3QC:classA <NRf+>	86
QC3QC:classA?	86
QC3QC:classB <NRf+>	87
QC3QC:classB?	87
PE1PE INIT	87
PE1PE:VOLT <CPD>	87
PE1PE:VOLT?	88
PE2PE INIT	88
PE2PE:VOLT <NRf+>	88
PE2PE:VOLT?	88
PD2PD INIT	89
PD2PD:VOLT <NR2>	89
PD2PD:VOLT?	89
PD2PD:PDO:LIST:VOLT?	89
PD2PD:PDO:LIST:CURR?	90
PD2PD:CC SWIT	90
PD2PD:CC?	90
PD3PD INIT	90
PD3PD:FIXED:VOLT <NR2>	91
PD3PD:FIXED:VOLT?	91
PD3PD:PPSn:VOLT <Nr1>,<Nrf+>	91
PD3PD:PPSn:VOLT? <NR1>	92
PD3PD:PDO:LIST:VOLT?	92
PD3PD:PDO:LIST:CURR?	92
PD3PD:CC SWIT	92
PD3PD:CC?	92
FCP INIT	93
FCP:VOLT <Nrf+>	93
FCP:VOLT?	93
FCP:LIST:VOLT?	93
FCP:LIST:CURR?	94
FCP:LIST:POW?	94
SCP INIT	94
SCP:VOLT <NRf+>	94
SCP:VOLT?	94
SCP:CURR <NRf+>	95
SCP:CURR?	95
SCP:LIST:VOLT?	95
SCP:LIST:CURR?	96
SCP:LIST:POW?	96
DPDN INIT	96
DPDN DPSH	96
DPDN DNSH	96
DPDN:VOLT?	96
QC:DISConnect <BOOL>	97
Chapter 12 Auto-test Commands	98
PROGram:MAXimum:CURREnt <NRf+>	98
PROGram:MAXimum:CURREnt?	98
PROGram:MAXimum:VOLTage <NRf+>	98
PROGram:MAXimum:VOLTage?	99
PROGram:MAXimum:POWEr <NRf+>	99

PROGrama:MAXimum:POWEr?	99
PROGrama:AUTO <BOOL>.....	100
PROGrama:AUTO?.....	100
PROGrama:START:VOLTage <NRf+>	100
PROGrama:START:VOLTage?.....	101
PROGrama:BEEP <CRD>	101
PROGrama:BEEP?	101
PROGrama:OUTPut:CONDition <CRD>	101
PROGrama:OUTPut:CONDition?.....	102
PROGrama:OUTPut:MODE <CRD>	102
PROGrama:OUTPut:MODE?	102
PROGrama:STOP:CONDition <CRD>.....	103
PROGrama:STOP:CONDition?	103
PROGrama:STEP:COUNT <NRf+>	103
PROGrama:STEP:COUNT?	104
PROGrama[:STEP]:QCModule:PROToCol <NR1>,<CRD>	104
PROGrama[:STEP]:QCModule:PROToCol?<NR1>	104
PROGrama[:STEP]:QCModule:VSRC <NR1>,<NRf+>	105
PROGrama[:STEP]:QCModule:VSRC? <NR1>.....	105
PROGrama[:STEP]:QCModule:ISRC <NR1>,<NRf+>	105
PROGrama[:STEP]:QCModule:ISRC? <NR1>	106
PROGrama[:STEP]:QCModule:POS <NR1>,<NR1>.....	106
PROGrama[:STEP]:QCModule:POS? <NR1>	106
PROGrama[:STEP]:MODE <NR1>,<CRD>	107
PROGrama[:STEP]:MODE? <NR1>	107
PROGrama[:STEP]:SHORt <NR1>,<bool>	107
PROGrama[:STEP]:SHORt? <NR1>	108
PROGrama[:STEP]:LEVel <NR1>,<NRf+>	108
PROGrama[:STEP]:LEVel? <NR1>	108
PROGrama[:STEP]:READback <NR1>,<CRD>	109
PROGrama[:STEP]:READback? <NR1>	109
PROGrama[:STEP]:READback:MAXimum <NR1>,<NRf+>	109
PROGrama[:STEP]:READback:MAXimum? <NR1>.....	110
PROGrama[:STEP]:READback:MINimum <NR1>,<NRf+>	110
PROGrama[:STEP]:READback:MINimum? <NR1>.....	110
PROGrama[:STEP]:DELay <NR1>,<NRf+>.....	111
PROGrama[:STEP]:DELay? <NR1>	111
PROGrama:SAVE <NR1>.....	111
PROGrama:RECall <NR1>	112
PROGrama[:STATe] <BOOL>	112
PROGrama[:STATe]?.....	112
PROGrama:RUN:STEP?.....	113
PROGrama[:STEP]:RESult? <NR1>.....	113
Chapter 13 Battery Commands.....	114
BATTery:DISCharge:CURRent:LIMit <NRf+>	114
BATTery:DISCharge:CURRent:LIMit?	114
BATTery:DISCharge:CURRent <NRf+>	114
BATTery:DISCharge:CURRent?.....	115
BATTery:STOP:VOLTage <NRf+>	115
BATTery:STOP:VOLTage?.....	115
BATTery:STOP:CAPacity <NRf+>	116
BATTery:STOP:CAPacity?.....	116
BATTery:STOP:TIME <NRf+>	116
BATTery:STOP:TIME?.....	117
BATTery:SAVE <NRf+>	117
BATTery:RECall <NRf+>	117
BATTery[:STATe] <BOOL>.....	118
BATTery[:STATe]?.....	118
BATTery:TIME?.....	118
BATTery:CAPacity?	118
Chapter 14 OCP Test Commands	120

OCP:VON <NRf+>.....	120
OCP:VON?	120
OCP:VON:DElAy <NRf+>.....	120
OCP:VON:DElAy?	121
OCP:CURRent:Range <NRf+>	121
OCP:CURRent:Range?	121
OCP:IStart <NRf+>	122
OCP:IStart?	122
OCP:IEND <NRf+>.....	122
OCP:IEND?	123
OCP:STEP <NRf+>	123
OCP:STEP?.....	123
OCP:DWELI <NRf+>	124
OCP:DWELI?.....	124
OCP:VTRig <NRf+>	124
OCP:VTRig?	125
OCP:MIN:TRIP <NRf+>.....	125
OCP:MIN:TRIP?	125
OCP:MAX:TRIP <NRf+>	125
OCP:MAX:TRIP?	126
OCP:SAV <NR1>.....	126
OCP:RCL <NR1>	127
OCP[:STATe] <BOOL>	127
OCP[:STATe]?.....	127
OCP:RESult[:OCP]?	128
OCP:RESult:PMAx?	128
Chapter 15 OPP Test Commands.....	129
OPP:VON <NRf+>	129
OPP:VON?	129
OPP:VON:DElAy <NRf+>	129
OPP:VON:DElAy?	130
OPP:CURRent:Range <NRf+>	130
OPP:CURRent:Range?	130
OPP:PStArt <NRf+>.....	131
OPP:PStArt?	131
OPP:PEND <NRf+>.....	131
OPP:PEND?	132
OPP:STEP <NRf+>	132
OPP:STEP?.....	132
OPP:DWELI <NRf+>	133
OPP:DWELI?.....	133
OPP:VTRig <NRf+>.....	133
OPP:VTRig?	134
OPP:MIN:TRIP <NRf+>.....	134
OPP:MIN:TRIP?	134
OPP:MAX:TRIP <NRf+>.....	134
OPP:MAX:TRIP?	135
OPP:SAV <NR1>.....	135
OPP:RCL <NR1>.....	135
OPP[:STATe] <BOOL>.....	136
OPP[:STATe]?.....	136
OPP:RESult[:OPP]?	136
OPP:RESult:PMAx?.....	137
Chapter 16 Trace Commands.....	138
TRACe:CLear.....	138
TRACe:FREE?	138
TRACe:POINts <NRf+>.....	138
TRACe:POINts?	138
TRACe:FEED[:SElEcted] <CRD>.....	139
TRACe:FEED[:SElEcted]?.....	139

TRACe:FEED:CONTRol <CRD>	139
TRACe:FEED:CONTRol?	140
TRACe:FILTer[:STATe] <BOOL>	140
TRACe:FILTer[:STATe]?	140
TRACe:DELay <NRf+>.....	141
TRACe:DELay?	141
TRACe:TIMer <NRf+>	141
TRACe:TIMer?.....	142
TRACe:DATA?.....	142
Chapter 17 Peak Commands.....	143
PEAK[:STATe] <bool>.....	143
PEAK CLear	143
PEAK:VOLTage:MAXimum?	143
PEAK:VOLTage:MINimum?	144
PEAK:CURRent:MAXimum?	144
PEAK:CURRent:MINimum?	144
Chapter 18 CR-LED Commands.....	145
LED:VOLTage	145
LED:CURRent	146
LED:RCOeff.....	146
Chapter 19 IEEE-488 Commands.....	147
*CLS - Clear Status	147
*ESE <NRf> - Event Enable	147
*ESE?	148
*ESR?	148
*IDN?	148
*OPC.....	149
*OPC?.....	150
*PSC	150
*PSC?	150
*RCL	151
*RST	151
*SAV	151
*SRE	152
*SRE?	152
*STB?	153
*TST?.....	153
*TRG.....	153
Chapter 20 Error Messages.....	155

Chapter 1 Remote Control

1.1 Overview

This chapter will provide following remote configuration introductions:

- SCPI Command Introduction
- Command type
- Command format
- Data format
- Remote Operation

1.2 SCPI Command Introduction

SCPI is short for Standard Commands for Programmable Instruments which defines a communication method of bus controller and instrument. It is based on ASCII and supply for testing and measuring instruments. SCPI command is based on hierarchical architecture which also known as tree system. In this system, Relevant Command is returned to a common node or root, so that a subsystem is formed.

A part of OUTPUT subsystem is listed below:

OUTPUT:

SYNC {OFF|0|ON|1}

SYNC:

MODE {NORMAL|CARRIER}

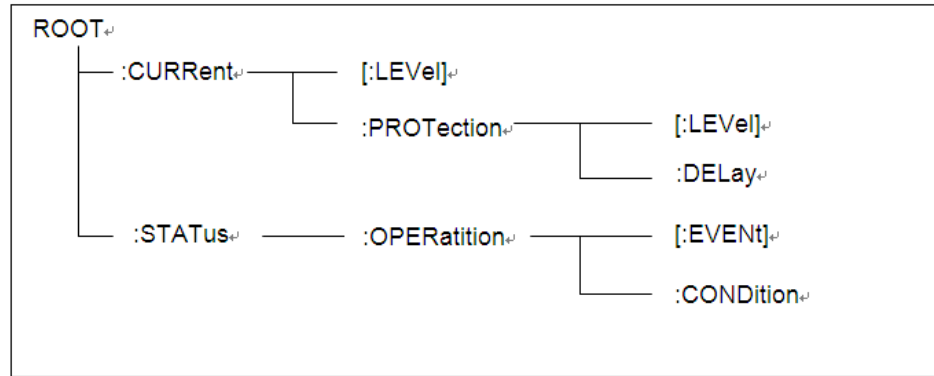
POLARITY {NORMAL|INVERTED}

OUTPUT is the root class keyword, SYNC is the second keyword, MODE and POLARITY are the third keyword. Colon(:) is used for separating the command keyword and the next level keyword.

1.3 Command Type of SCPI

SCPI has two types of commands, common and subsystem.

- Common commands generally are not related to specific operation but to controlling overall electronic load functions, such as reset, status, and synchronization. All common commands consist of a three-letter mnemonic preceded by an asterisk: *RST *IDN? *SRE 8.
- Subsystem commands perform specific electronic load functions. They are organized into an inverted tree structure with the "root" at the top. The following figure shows a portion of a subsystem command tree, from which you access the commands located along the various paths.



Multiple commands in a message

Multiple SCPI commands can be combined and sent as a single message with one message terminator. There are two important considerations when sending several commands within a single message:

- Use a semicolon to separate commands within a message.
- Head paths influence how the instrument interprets commands.

We consider the head path as a string which will be inserted in front of every command of a message. As for the first command of a message, the head path is a null string; for each subsequent command, the head path is a string which is defined to form the current command until and including the head of the last colon separator. A message with two combined commands: `CURR:LEV 3;PROT:STAT OFF`

The example indicates the effect of semicolon and explains the concept of head path. Since the head path is defined to be "CURR" after "curr: lev 3", the head of the second command, "curr", is deleted and the instrument explains the second command as: `CURR:PROT:STAT OFF`

If "curr" is explicitly included in the second command, it is semantically wrong. Since combining it with the head path will become "CURR:CURR:PROT:STAT OFF", resulting in wrong command.

Movement in the subsystem

In order to combine commands from different subsystems, you need to be able to reset the header path to a null string within a message. You do this by beginning the command with a colon (:), which discards any previous header path. For example, you could clear the output protection and check the status of the Operation Condition register in one message by using a root specifier as follows:

```
PROTection:CLEAr;:STATus:OPERation:CONDition?
```

The following message shows how to combine commands from different subsystems as well as within the same subsystem:

```
POWER:LEVel 200;PROTection 28; :CURRent:LEVel 3;PROTection:STATe ON
```

Note the use of the optional header `LEVel` to maintain the correct path within the voltage and current subsystems, and the use of the root specifier to move between subsystems.

Including Common Commands

You can combine common commands with subsystem commands in the same message. Treat the common command as a message unit by separating it with a semicolon (the message unit separator). Common commands do not affect the header path; you may insert them anywhere in the message.

```
VOLTage:TRIGgered 17.5;:INITialize;*TRG
```

```
OUTPut OFF;*RCL 2;OUTPut ONIT872X-3X SCPI Communication protocol 17
```

Case sensitivity

Common commands and SCPI commands are not case sensitive. You can use upper or lower, for example:

```
*RST = *rst  
:DATA? = :data?  
:SYSTem:PRESet = :system:preset
```

Long-form and short-form versions

A SCPI command word can be sent in its long-form or short-form version. However, the short-form version is indicated by upper case characters. Examples:

```
:SYSTem:PRESet long-form
```

```
:SYST:PRES short form
```

```
:SYSTem:PRES long-form and short-form combination
```

Note that each command word must be in long-form or short-form, and not something in between.

For example, :SYSTe:PRESe is illegal and will generate an error. The command will not be executed.

Query

Observe the following precautions with queries:

- Set up the proper number of variables for the returned data. For example, if you are reading back a measurement array, you must dimension the array according to the number of measurements that you have placed in the measurement buffer.
- Read back all the results of a query before sending another command to the electronic load. Otherwise a Query Interrupted error will occur and the unreturned data will be lost.

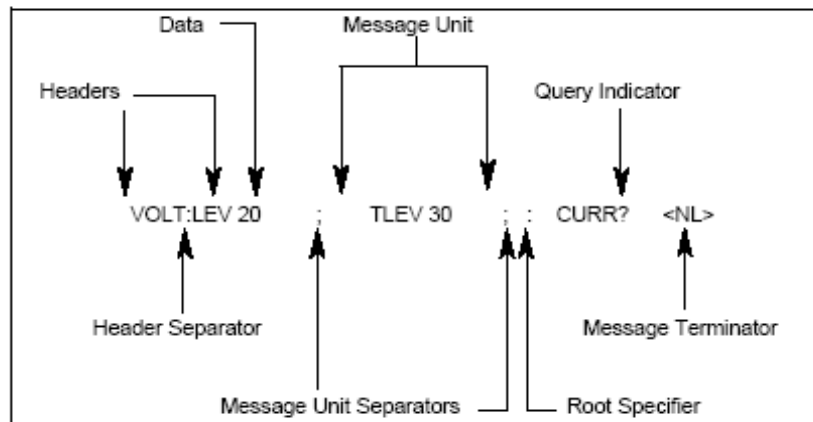
1.4 Message Type of SCPI

There are two types of SCPI messages, program and response.

- program message: A program message consists of one or more properly formatted SCPI commands sent from the controller to the electronic load. The message, which may be sent at any time, requests the electronic load to perform some action.
- response message: A response message consists of data in a specific

SCPI format sent from the electronic load to the controller. The electronic load sends the message only when commanded by a program message called a "query."

The next figure illustrates SCPI message structure:



The Message Unit

The simplest SCPI command is a single message unit consisting of a command header (or keyword) followed by a message terminator. The message unit may include a parameter after the header. The parameter can be numeric or a string.

VOLTage 20<NL>

Headers

Headers, also referred to as keywords, are instructions recognized by the electronic load. Headers may be either in the long form or the short form. In the long form, the header is completely spelled out, such as VOLTAGE, STATUS, and DELAY. In the short form, the header has only the first three or four letters, such as VOLT, STAT, and DEL.

Query Indicator

Following a header with a question mark turns it into a query (VOLTage?, VOLTage:PROTection?). If a query contains a parameter, place the query indicator at the end of the last header (VOLTage:PROTection?MAX).

Message Unit Separator

When two or more message units are combined into a compound message, separate the units with a semicolon (STATus:OPERation?;QUEStionable?).

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.

Message Terminator

A terminator informs SCPI that it has reached the end of a message. Three permitted message terminators are:

- newline (<NL>), decimal 10 or hexadecimal 0X0A in ASCII.

- end or identify (<END>)
- both of the above (<NL><END>).

In the examples of this guide, there is an assumed message terminator at the end of each message.

Command execution rules

- Commands execute in the order that they are presented in the program message.
- An invalid command generates an error and, of course, is not executed.
- Valid commands that precede an invalid command in a multiple command program message are executed.
- Valid commands that follow an invalid command in a multiple command program message are ignored.

1.5 Response Data Type

Character strings returned by query statements may take either of the following forms, depending on the length of the returned string:

<CRD>	Character Response Data. Permits the return of character strings.
<AARD>	Arbitrary ASCII Response Data. Permits the return of un delimited 7-bit ASCII. This data type has an implied message terminator.
<SRD>	String Response Data. Returns string parameters enclosed in double quotes

Response messages

A response message is the message sent by the instrument to the computer in response to a query command.

Sending a response message

After sending a query command, the response message is placed in the Output Queue. When the IT8500+ Series is then addressed to talk, the response message is sent from the Output Queue to the computer.

Multiple response messages

If you send more than one query command in the same program message (see the paragraph entitled, "Multiple Command Messages"), the multiple response messages for all the queries is sent to the computer when the IT8500+ Series is addressed to talk. The responses are sent in the order that the query commands were sent and are separated by semicolons (;). Items within the same query are separated by commas (,). The following example shows the response message for a program message that contains four single item query commands:

0; 1; 1; 0

Response message terminator (RMT)

Each response is terminated with an LF (line feed) and EOI (end or identify). The following example shows how a multiple response message is terminated:

0; 1; 1; 0; <RMT>

Message exchange protocol

Two rules summarize the message exchange protocol:

Rule 1. You must always tell the IT8500+ Series what to send to the computer.

The following two steps must always be performed to send information from the instrument other computer:

1. Send the appropriate query command(s) in a program message.
2. Address the IT8500+ Series to talk.

Rule 2. The complete response message must be received by the computer before another program message can be sent to the IT8500+ Series.

1.6 Command Format

Formats for command display are as follows:

```
[SOURce[1|2]:]VOLTage:UNIT {VPP|VRMS|DBM}
```

```
[SOURce[1|2]:]FREQuency:CENTer  
{<frequency>|MINimum|MAXimum|DEFault}
```

Based on the command syntax, most commands (and certain Parameter) are expressed in both upper and lower cases. Upper case refers to abbreviation of commands. Shorter program line may send commands in abbreviated format. Long-format commands may be sent to ensure better program readability.

For example, both formats of VOLT and VOLTAGE are acceptable in the above syntax statements. Upper or lower case may be used. Therefore, formats of VOLTAGE, volt and Volt are all acceptable. Other formats (such as VOL and VOLTAG) are invalid and will cause errors.

- Parameter options with given command strings are included in the brace ({}). The brace is not sent along with command strings.
- Vertical stripes (|) separate several parameter options with given command strings. For example, {VPP|VRMS|DBM} indicates that you may assign "APP", "VRMS" or "DBM" in the above commands. Vertical stripes are not sent along with command strings.
- Angle brackets (< >) in the second example indicates that a value must be assigned to the parameter in the brace. For example, the parameter in the angle bracket is <frequency> in the above syntax statements. Angle brackets are not sent along with command strings. You must assign a value (such as "FREQ:CENT 1000") to the parameter, unless you select other options displayed in the syntax (such as "FREQ:CENT MIN").
- Some syntax elements (such as nodes and Parameter) are included in square brackets ([]). It indicates that these elements can be selected and omitted. Angle brackets are not sent along with command strings. If no value is assigned to the optional Parameter, the instrument will select a default value. In the above examples, "SOURce[1|2]" indicates that you may refer to source channel 1 by "SOURce" or "SOURce1" or "SOUR1" or "SOUR". In addition, since the whole SOURce node is optional (in the square bracket), you can refer to the channel 1 by omitting the whole SOURce node. It is because the channel 1 is the default channel for SOURce language node. On the other hand, if you want to refer to channel 2, "SOURce2" or "SOUR2" must be used in the program line.

Colon (:)

It is used to separate key words of a command with the key words in next level. As shown below:

```
APPL:SIN 455E3,1.15,0.0
```

In this example, APPLy command assigns a sine wave with frequency of 455 KHz, amplitude of 1.15 V and DC offset of 0.0 V.

Semicolon (;)

It is used to separate several commands in the same subsystem and can also minimize typing. For example, to send the following command string:

```
TRIG:SOUR EXT; COUNT 10
```

has the same effect as sending the following two commands:

```
TRIG:SOUR EXT  
TRIG:COUNT 10
```

Question mark (?)

You can insert question marks into a command to query current values of most Parameter. For example, the following commands will trigger to set the count as 10:

```
TRIG:COUN 10
```

Then, you may query count value by sending the following command:

```
TRIG:COUN?
```

You may also query the allowable minimum or maximum count as follows:

```
TRIG:COUN?MIN  
TRIG:COUN?MAX
```

Comma (,)

If a command requires several Parameter, then a comma must be used to separate adjacent Parameter.

Space

You must use blank characters, [TAB] or [Space] to separate Parameter with key words of commands.

Generic commands (*)

Execute functions like reset, self inspection and status operation. Generic commands always start with a asterisk (*) and occupy 3 character sizes, including one or more Parameter. Key words of a command and the first parameter are separated by a space. Semicolon (;) can separate several commands as follows:

```
*RST; *CLS; *ESE 32; *OPC?
```

Command terminator

Command strings sent to the instrument must end with a <Newline> (<NL>) character. IEEE-488 EOI (End or Identify) information can be used as <NL> character to replace termination command string of <NL> character. It is acceptable to place one <NL> after a <Enter>. Termination of command string

always resets current SCPI command path to root level.



Note

As for every SCPI message with one query sent to the instrument, the instrument will use a <NL> or newline sign (EOI) to terminate response of return. For example, if "DISP:TEXT?" is sent, <NL> will be placed after the returned data string to terminate response. If an SCPI message includes several queries separated by semicolon (such as "DISP?;DISP:TEXT?"), <NL> will terminate response returned after response to the last query. In all cases, the program must read <NL> in response before another command is sent to the instrument, otherwise errors will be caused.

1.7 Data Type

SCPI language defines several data types used for program message and response messages.

- Numerical parameter

Commands requiring numerical Parameter support the notations of all common decimal notations, including optional signs, decimal points, scientific notation, etc. Special values of numerical Parameter are also acceptable, such as MIN, MAX and DEF. In addition, suffixes for engineering units can also be sent together with numerical Parameter (including M, k, m or u). If the command accepts only some specific values, the instrument will automatically round the input Parameter to acceptable values. The following commands require numerical Parameter of frequency value:

[SOURce[1|2]:]FREQUency:CENTer {<Frequency>|MINimum|MAXimum}

- Discrete parameter

Discrete Parameter are used for settings with limited number of programming values (such as IMMEDIATE, EXTERNAL or BUS). They can use short and long format like key words of commands. They may be expressed in both upper and lower case. The query response always returns uppercase Parameter in short format. The following commands require discrete Parameter in voltage unit:

[SOURce[1|2]:]VOLTage:UNIT {VPP|VRMS|DBM}

- Boolean parameter

Boolean Parameter refer to true or false binary conditions. In case of false conditions, the instrument will accept "OFF" or "0". In case of true conditions, the instrument will accept "ON" or "1". In query of Boolean settings, the instrument will always return "0" or "1". Boolean Parameter are required by the following commands:

DISPlay {OFF|0|ON|1}

- ASCII string Parameter

String Parameter may actually include all ASCII character sets. Character strings must start and end with paired quotation marks; and single quotation marks or double quotation marks are both allowed. Quotation mark separators may also act as one part of a string, they can be typed twice without any character added between them. String parameter is used in the following command:

DISPlay:TEXT <quoted string>

For example, the following commands display message of "WAITING..." (without quotation marks) on the front panel of the instrument.

DISP:TEXT "WAITING..."

Single quotation marks may also be used to display the same message.

DISP:TEXT 'WAITING...'

1.8 Remote Interface Communication

The rear panel of this series of loads is equipped with a USB interface as standard, and supports two modes: TMC and VCP. Before connecting to the PC, please select TMC or VCP through the front panel. If you choose the VCP mode, after connecting to the PC, you need to ensure that the serial port parameter settings on the PC software interface are consistent with the load device (9600/8/N/1).

USB Interface

Use a cable with USB interfaces on both ends (one end is USB A type, the other end is USB B type interface) to connect the load and the computer. All electronic load functions can be programmed via USB.

After connecting the load to the PC via USB, if you choose the VCP mode, you need to install the ITECH driver (download it on the official website or contact the ITECH technical support staff to obtain it), otherwise the communication will not succeed.

LAN Interface

In addition to the USB interface, IT8512G+ and IT8512BG+ models also have a LAN communication interface as standard.

Before connecting the device and PC, please set the following parameters:

1. Press Shift+8 to enter the System menu.
2. Press the left and right direction keys or the knob to select COMM and press Enter.
3. Press the left and right direction keys or the knob to select LAN and press Enter.
4. Set the IP address and press Enter.
The IP address of the device must be in the same network segment as the IP address of the PC.
5. Set the MK subnet mask and press Enter.
6. Set the gateway address and press Enter.
7. Set the port number (2000~65535) and press Enter.

Chapter 2 SCPI Status Register

SCPI protocol supports the following four groups of registers.

Questionable Status Register Group

Questionable status register group includes three 16-bit registers: status register, event register and enable register. When the corresponding status register bit is changed, the corresponding bit in the event register is set. If the corresponding bit in the enable register bit is set, it will be generated once Event (status byte register QUES is set). After executing an event register read operation, the event register will be automatically cleared. The status register is defined as follows:

Bit	Signal	Description
Bit0	VF	Either an overvoltage or a reverse voltage has occurred This bit reflects the active state of the FLT pin on the back of the unit. The bit remains set until the condition is removed and PROT:CLE is programmed.
Bit1	OC	An over-current condition has occurred.
Bit3	OP	An overpower condition has occurred.
Bit4	OT	An over-temperature condition has occurred.
Bit8	SV	Remote measurement terminal is not connected.
Bit11	UNR	The input is unregulated, when the input is regulated the bit is cleared.
Bit13	OV	An over voltage condition has occurred. Both this bit and VF bit0 are set and the loads are turned off. Both bits remain set until the condition is removed and PROT:CLE is programmed.

Standard Event Status Register Group

The standard event register group consists of two 16-bit registers: the event register and the enable register. When an event occurs, and if the corresponding bit in the enable register is set, an event occurs (the ESB in the Status Byte register is set). After executing an event register read operation, the event register will be automatically cleared. The event register is defined as follows:

Bit	Signal	Description
Bit0	OPC	The load has completed all pending operations. *OPC must be programmed for this bit to be set when pending operations are complete.
Bit2	QYE	The output queue was read with no data present or the data was lost. Errors in the range of 499 through 400 can set this bit.
Bit3	DDE	Device-Dependent Error. Memory was lost or self-test failed. Errors in the range of 399 through 300 can set this bit.
Bit4	EXE	A command parameter was outside its legal range, inconsistent with the load's operation, or prevented from executing because of an operating condition. Errors in the range of 299 through 200 can set this bit.
Bit5	CME	A syntax or semantic error has occurred or the load received a <get> within a program message. Errors in the range of 199 through 100 can set this bit.

Bit7	PON	The unit has been turned off and then on since this bit was last read.
------	-----	--

Operation Status Register Group

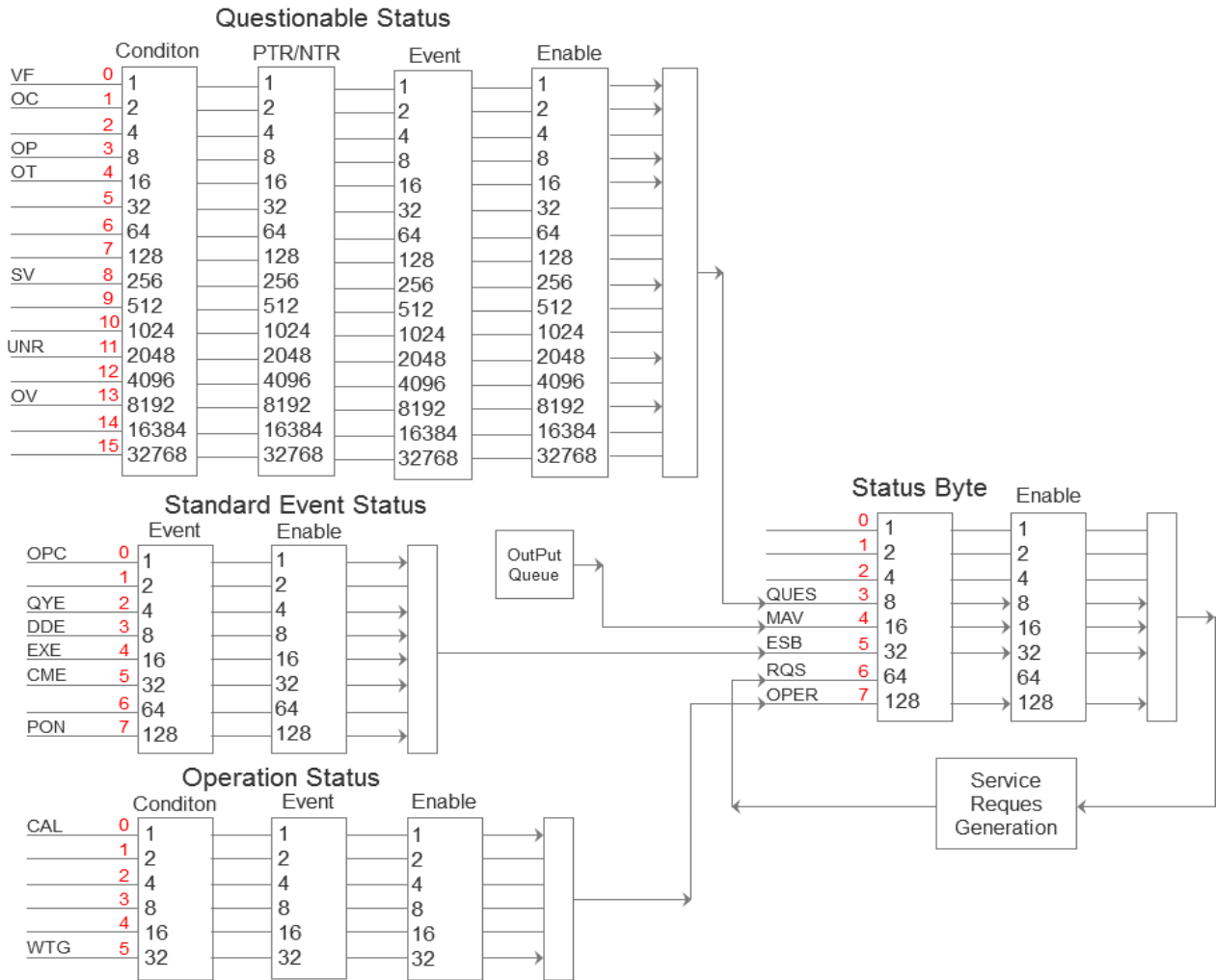
The operational status register group consists of three 16-bit registers: status register, event register, and enable register. When the status register corresponding bit is changed, the corresponding event register bit will be set. If the corresponding bit in the enable register bit is set, it will be generated once Event (status byte register OPER is set). After executing an event register read operation, the event register will be automatically cleared. The status register is defined as follows:

Bit	Signal	Description
Bit0	CAL	The electronic load iscalculated a new calibration constant.
Bit5	WTG	The electronic load is waiting for a trigger.

Status Byte Register Group

The status byte register group consists of two 8-bit registers: event register and enable register. If the corresponding bit in the enable register bit is set, it will be generated once Event (status byte register RQS is set). The status byte register will be automatically cleared when an event register read is executed. The status register is defined as follows:

Bit	Signal	Description
Bit3	QUES	Indicates if an enabled questionable event has occurred.
Bit4	MAV	Indicates if the Output Queue contains data.
Bit5	ESB	Indicates if an enabled standard event has occurred.
Bit6	RQS	During a serial poll, RQS is returned and cleared.
Bit7	OPER	Indicates if an operation event has occurred.



Chapter 3 System Commands

SYSTem:POSetup

This command is used to set the state of the load when it is powered on, including the working mode (CC/CV/CW/CR) and set value of the load.

RST: restore factory settings

SAV0: Recall the parameter settings saved in the 0th group

Command Syntax

SYSTem:POSetup

Parameters

RST|SAV0

Returned Parameters

Not involved

Examples

SYST:POS SAV0

SYSTem:POSetup?

This command is used to query the state of the load when it is powered on.

Command Syntax

SYSTem:POSetup?

Parameters

None

Returned Parameters

<CRD>

SYSTem:BEEPer:IMMediate

This command is used to test the buzzer and make it beep once.

Command Syntax

SYSTem:BEEPer:IMMediate

Parameters

None

Returned Parameters

Not involved

Examples

SYST:BEEP:IMM

SYSTem:MEMory

This command is used to specify the group name for data recall. It is equivalent to the parameter setting of Memory Group in the System menu.

Command Syntax

SYSTem:MEMory

Parameters

<NR1>

0~9

Returned Parameters

Not involved

Examples

SYST:MEM 2

SYSTem:MEMory?

This command is used to query the group name.

Command Syntax

SYSTem:MEMory?

Parameters

None

Returned Parameters

<NR1>

SYSTem:COMMunicate:SElect

The command sets the communication mode: TMC, VCP and LAN. Different models support different communication interfaces.

The actual model is subject to the specific model.

Command Syntax

SYSTem:COMMunicate:SElect

Parameters

<CRD>

USBVCP|USBTMC|LAN

Returned Parameters

Not involved

Examples

SYST:COMM:SEL LAN

SYSTem:COMMunicate:SElect?

This command is used to query the communication mode.

Command Syntax

SYSTem:COMMunicate:SElect?

Parameters

None

Returned Parameters

<CRD>

SYSTem:RUNMode

This command is used to set the operating mode of the load.

NORMal: Normal test mode (CC/CV/CW/CR)

BATTery: battery test mode

PROGram: automatic test mode

OCP: OCP test mode

OPP: OPP test mode

Command Syntax

SYSTem:RUNMode

Parameters

<CRD>

NORMal|BATTery|PROGram|OCP|OPP

Returned Parameters

Not involved

Examples

SYSTem:RUNMode OPP

SYSTem:RUNMode?

This command is used to query the current operating mode.

Command Syntax

SYSTem:RUNMode?

Parameters

None

Returned Parameters

<CRD>

SYSTem:PRESet

This command is used to restore to factory settings.

Command Syntax

SYSTem:PRESet

Parameters

None

Returned Parameters

Not involved

Examples

SYSTem:PRESet

SYSTem:DEFault

This command is used to restore to factory settings.

Command Syntax

SYSTem:DEFault

Parameters

None

Returned Parameters

Not involved

Examples

SYSTem:DEFault 1

SYSTem:CLEar

This command is used to clear the error sequence.

Command Syntax

SYSTem:CLEar

Parameters

None

Returned Parameters

Not involved

Examples

SYSTem:CLEar

SYSTem:ERRor[:NEXT]?

This command return the next error number, followed by a remote programming error message string.

Sequence is a FIFO buffer FIFO (first-in, first-out), when the error occurs, the error is stored in the cache. When it is read out, it is deleted from the sequence.

After reading all the errors, the query Returned "0, No Error". If the error accumulates too much that is more than the cache can bear, the last error of the sequence will be "-350, Too Many Errors".

Command Syntax

SYSTem:ERRor[:NEXT]?

Parameters

None

Returned Parameters

<NR1>, <SRD>

Examples

SYSTem:ERRor[:NEXT]?

SYSTem:VERSion?

This query returns the SCPI revision of the load used. The format is YYYY.V, where YYYY is the year and V is the revision number for that year.

Command Syntax

SYSTem:VERSion?

Parameters

None

Returned Parameters

<NR1>, <SRD>

Examples

SYSTem:VERS?

SYSTem:SENSe[:STATe] <bool>

This command is used to control the remote compensation function on and off.

Command Syntax

SYSTem:SENSe[:STATe] <bool>

Parameters

<bool>

0|1|OFF|ON

Examples

SYST:SENS ON

Query Syntax

SYSTem:SENSe[:STATe]?

Returned Parameters

0|1

*RST Value

OFF

SYSTem:SENSe[:STATe]?

This command is used to query the status of remote compensation.

Command Syntax

SYSTem:SENSe[:STATe]?

Parameters

Bool

Returned Parameters

<NR1>

SYSTem:LOCal

The command sets the load to local mode. Can use the buttons on front panel to operate.

Command Syntax

SYSTem:LOCal

Parameters

None

Examples

SYST:LOC

SYSTem:REMote

This command is used to put the load in remote mode. Disable all the keys on the front panel except the local key and the Shift key. In the remote state, press the LOCAL key to return to the local mode.

Command Syntax

SYSTem:REMote

Parameters

None

Examples

SYST:REM

SYSTem:RWLock

This command is used to lock the panel keys of the load (except the Shift and On/Off keys). If you need to unlock, you can use the panel key combination: first press Shift, then press On/Off.

If you execute the "SYSTem:REMote" command first, and then execute the "SYSTem:RWLock" command, you cannot switch to the local mode through the Local button on the front panel, and you cannot use the panel combination keys: first press Shift, then press On/Off Way to unlock. Only through the SYSTem:LOCal instruction to return to the local mode, and then through the panel combination of keys: first press Shift, then press On/Off to unlock.

Command Syntax

SYSTem:RWLock

Parameters

None

Examples

SYST:RWL

Related Commands

SYST:REM

SYST:LOC

Chapter 4 Multi-channel commands

SYSTem:CHANnel:NUMBer

This command is used to set the channel number of the load. Setting range: 1~16.

Command Syntax

SYSTem:CHANnel:NUMBer

Parameters

<NR1>
1~16

Returned Parameters

Not involved

Examples

SYSTem:CHANnel:NUMBer 2

SYSTem:CHANnel:NUMBer?

This command is used to query the channel number.

Command Syntax

SYSTem:CHANnel:NUMBer?

Parameters

None

Returned Parameters

<NR1>

LINK:MODE

Set and query the status of Link.

This command is used to set the load synchronization mode when multiple channels are connected.

OFF: Turn off the synchronization function

SYNChronous: Only On/Off synchronization

DUPLicate: Synchronization of On/Off, working mode and related parameter settings

Command Syntax

LINK:MODE

Parameters

<CRD>

OFF|SYNChronous|DUPLicate

Returned Parameters

Not involved

Examples

LINK:MODE DUPL

LINK:MODE?

This command is used to query the synchronization status.

Command Syntax

LINK:MODE?

Parameters

None

Returned Parameters

<CRD>

CHANnel

Set the default channel number for PI communication.

This command is used to specify the instrument of the corresponding channel for communication with the PC. For example, in the multi-channel function, by first sending this instruction to specify the channel number, the subsequent programming instructions will only take effect for the channel device, which facilitates the writing of subsequent instructions without specifying the corresponding channel number in each programming instruction.

Setting range: 1~16.

Command Syntax

CHANnel

Parameters

<NR1>

Range: 1~16

Returned Parameters

Not involved

Examples

CHANnel 5

CHANnel?

This command is used to query the channel number.

Command Syntax

CHANnel?

Parameters

None

Returned Parameters

<NR1>

Chapter 5 Status Commands

Those commands configure the status registers of the electronic load.

STATus:QUEStionable[:EVENT]?

This query returns the value of event register. Event register is read only register, it keeps all events sent to it. Read the quest event register will clear it.

Query Syntax

STATus:QUEStionable[:EVENT]?

Parameters

None

Examples

STAT:QUES:EVEN?

Returned Parameters

<NR1>

Related Commands

*CLS

STATus:QUEStionable:CONDition?

This command can read the parameter from quest condition register. It is a read only register, keep the real-time(not locked) query status of the load.

Query Syntax

STATus:QUEStionable:CONDition?

Parameters

None

Examples

STAT:QUES:COND?

Returned Parameters

<NR1>

Related Commands

STAT:OPER:COND?

STATus:QUEStionable:ENABLE <NR1>

This command is used to set the value of the enable register.

Command Syntax

STATus:QUEStionable:ENABLE <NR1>

Parameters

0 ~ 65535

Examples

STAT:QUES:ENAB 32

Query Syntax

STATus:QUEStionable:ENABLE?

Returned Parameters

<NR1>

STATus:QUEStionable:ENABLE?

This command is used to query the value of enable register.

Command Syntax

STATus:QUEStionable:ENABLE?

Parameters

None

Returned Parameters

<NR1>

STATus:OPERation[:EVENT]?

This command query the query operation event register values. The event register is read-only register, which holds (latches) all value passed by the NTR and, or PTR filter. Read channel operation event register will clear it.

Query Syntax

STATus:OPERation[:EVENT]?

Parameters

None

Examples

STAT:OPER:EVEN?

Returned Parameters

<NR1>

Related Commands

*CLS

STATus:OPERation:CONDition?

This query returns the value of operation condition register. That is a read-only register that holds the real-time (unlatched) operational status of the electronic load.

Query Syntax

STATus:OPERation:CONDition?

Parameters

None

Examples

STAT:OPER:COND?

Returned Parameters

<NR1>

Related Commands

STAT:QUES:COND?

STATus:OPERation:ENABLE <NR1>

The command is used to set the value of operations enable register. This register is a mask for enabling specific bits from the Operation Event register to set the operation summary bit (OPER) of the Status Byte register. The operation summary bit is the logical OR of all enabled Operation Event register bits.

Command Syntax

STATus:OPERation:ENABLE <NR1>

Parameters

<NR1>

0 - 65535

Default value

0

Examples

STAT:OPER:ENAB 32

Query Syntax

STATus:OPERation:ENABLE?

Returned Parameters

<NR1>

Related Commands

STAT:OPER?

STATus:OPERation:ENABLE?

The command is used to query the value of operations enable register. This register is a mask for enabling specific bits from the Operation Event register to set the operation summary bit (OPER) of the Status Byte register. The operation summary bit is the logical OR of all enabled Operation Event register bits.

Command Syntax

STATus:OPERation:ENABLE?

Parameters

None

Returned Parameters

<NR1>

Chapter 6 Input Settings

These commands control the input of the electronic load. The INPut, CURRent, RESistance and VOLTage commands program the actual input current, resistance, and voltage.

[SOURce:]INPut <bool>

This command is used to set the input state.

Command Syntax

[SOURce:]INPut <bool>

Parameters

<bool>

0 | 1 | OFF | ON

*RST Value

OFF

Examples

INP 1

Query Syntax

INPut[:STATe]?

Returned Parameters

0 | 1

[SOURce:]INPut?

This command is used to query the input status.

Command Syntax

[SOURce:]INPut?

Returned Parameters

0 | 1

Examples

INP?

[SOURce:]INPut:SHORT <bool>

This command is used to set the state of the short-circuit function.

Command Syntax

[SOURce:]INPut:SHORt <bool>

Parameters

0 | 1 | OFF | ON

*RST Value

OFF

Examples

INP:SHOR 1

Query Syntax

INPut:SHORt?

Returned Parameters

0 | 1

Related Commands

INP

[SOURce:]INPut:SHORt?

This command is used to query the short circuit status.

Command Syntax

[SOURce:]INPut:SHORt?

*RST Value

OFF

Examples

INP:SHOR?

[SOURce:]INPut:TIMer[:STATe] <bool>

This command is used to set whether the electronic load timer is turned on (corresponding to "TIMER" under the Config menu).

Command Syntax

[SOURce:]INPut:TIMer[:STATe] <bool>

Parameters

<bool>

0 | 1 | OFF | ON

*RST Value

OFF

Examples

INP:TIM 1

[SOURce:]INPut:TIMer[:STATe]?

This command is used to query whether the electronic load timer is on.

Command Syntax

[SOURce:]INPut:TIMer[:STATe]?

Examples

INP:TIM?

Returned Parameters

0 | 1 | OFF | ON

[SOURce:]INPut:TIMer:DELay <NRf+>

This command is used to set the timing time of the electronic load timer.

Command Syntax

[SOURce:]INPut:TIMer:DELay <NRf+>

Parameters

0.1 to 9999.9 | MINimum | MAXimum | DEFault (10)

Unit

S

Examples

[SOURce:]INPut:TIMer:DELay 20

[SOURce:]INPut:TIMer:DELay?

This command is used to query the timing time of the electronic load timer.

Command Syntax

[SOURce:]INPut:TIMer:DELay?

Returned Parameters

<NRf+>

[SOURce:]FUNction:MODE <CPD>

It is used to set whether the input mode is controlled by the list value or by the FUNCTION command setting.

FIXed: CC/CV/CW/CR mode

LIST: List mode

Command Syntax

[SOURce:]FUNCTION:MODE

Parameters

<CRD>

<FIXed | LIST>

Examples

[SOURce:]FUNCTION:MODE LIST

[SOURce:]FUNCTION:MODE?

Used to query the functional mode of the load.

Command Syntax

[SOURce:]FUNCTION:MODE?

Returned Parameters

<FIXed | LIST>

[SOURce:]FUNCTION <function>

[SOURce:]MODE <function>

These two commands are equivalent and are used to select the input mode of the load.

Command Syntax

[SOURce:]FUNCTION <function>

[SOURce:]MODE <function>

Parameters

<CRD>

CURRent|VOLTage|POWER|RESistance|LED|DYNamic|LIST

- CURRent: constant current operation mode
- VOLTage: constant voltage operation mode
- POWER: constant power operation mode
- RESistance: constant resistance operation mode
- DYNamic: dynamic operation mode
- LED: LED mode
- LIST: LIST mode

***RST Value**

CURRent

Examples

MODE RES

Query Syntax

[SOURce:]FUNction?

[SOURce:]MODE?

Returned Parameters

<CRD>

[SOURce:]FUNction?**[SOURce:]MODE?**

These two commands are equivalent and are used to query the input mode of the load.

Command Syntax

[SOURce:]FUNction?

[SOURce:]MODE?

Returned Parameters

CURRent|VOLTage|POWer|RESistance|LED|DYNamic|LIST

[SOURce:]TRANSient[:STATe] <BOOL>

Turn dynamic test mode on or off

Command Syntax

[SOURce:]TRANSient[:STATe] <BOOL>

Parameters

0 | 1 | OFF | ON

***RST Value**

Not involved

Examples

TRAN 0

Query Syntax

[SOURce:]TRANSient[:STATe]?

Returned Parameters

0 | 1

[SOURce:]TRANSient[:STATe]?

This command is used to view the status of the dynamic test mode.

Command Syntax

[SOURce:]TRANSient[:STATe]?

Parameters

None

Returned Parameters

0 | 1

OUTPut:INHibit:MODE <CRD>

Set the prohibited output mode of external IO

Command Syntax

OUTPut:INHibit:MODE < CRD >

Parameters

LIVING|LATCH

*RST Value

Not involved

Examples

OUTPut:INHibit:MODE LATCH

Query Syntax

OUTPut:INHibit:MODE?

Returned Parameters

LIVING|LATCH

OUTPut:INHibit:MODE?

This command queries the prohibited output mode of external IO.

Command Syntax

OUTPut:INHibit:MODE?

Parameters

None

Returned Parameters

LIVING|LATCH

INPut:SDS:SURGe:SUPPpress <BOOL>

Set the state of anti-surge function (RC-CTRL)

Command Syntax

INPut:SDS:SURGe:SUPPpress <BOOL>

Parameters

0|1|OFF|ON

*RST Value

OFF

Examples

INP:SDS:SURG:SUPP ON

Query Syntax

INPut:SDS:SURGe:SUPPpress?

Returned Parameters

0 | 1

INPut:SDS:SURGe:SUPPpress?

Query the status of anti-surge function (RC-CTRL)

Command Syntax

INPut:SDS:SURGe:SUPPpress?

Parameters

None

Returned Parameters

0 | 1

[SOURce:]CURRent:RANGe <NRf+>

This command is used to set the range of load current. When the set parameter is in the low range, select the low current range, otherwise select the high current range.

Command Syntax

[SOURce:]CURRent:RANGe <NRf+>

Parameters

<NRf+>

0 to rated value | MINimum | MAXimum | DEF

Unit

A

*RST Value

MAXimum (high range)

Examples

CURR:RANGe MIN

Query Syntax

[SOURce:]CURRent:RANGe?

Returned Parameters

<NR2>

[SOURce:]CURRent:RANGe?

This command is used to query the load current range.

Command Syntax

[SOURce:]CURRent:RANGe?

Parameters

None

Unit

A

*RST Value

MAXimum (high range)

[SOURce:]VOLTage:RANGe <NRf+>

This command is used to set the voltage range of the load. When the set parameter is in the low range, select the low voltage range, otherwise select the high voltage range.

Command Syntax

[SOURce:]VOLTage:RANGe <NRf+>

Parameters

0 to rated value | MINimum | MAXimum|DEF

Unit

V

*RST Value

MAXimum(high range)

Examples

SOUR:VOLT:RANG MIN

Query Syntax

[SOURce:]VOLTage:RANGe?

Returned Parameters

<NR2>

[SOURce:]VOLTage:RANGe?

This command is used to query the voltage range of the load.

Command Syntax

[SOURce:]VOLTage:RANGe?

Parameters

None

Unit

V

*RST Value

MAXimum(high range)

[SOURce:]VOLTage:RANGe:AUTO[:STATe]

This command sets the automatic range state of the voltmeter of the load.

Command Syntax

[SOURce:]VOLTage:RANGe:AUTO[:STATe] <bool>

Parameters

<BOOL>

0 | 1 | ON | OFF

Returned Parameters

0|1

Examples

VOLT:RANG:AUTO 1

Query Syntax

[SOURce:]VOLTage:RANGe:AUTO[:STATe]?

Returned Parameters

<NR1>

[SOURce:]CURRENT:SLEW[:BOTH] <NRf+>

This command is used to set the current rising and falling slope (A/uS).

Command Syntax

[SOURce:]CURRENT:SLEW[:BOTH] <NRf+>

Unit

A /uS

Parameters

MIN~MAX | MAX | MIN | DEF

Examples

CURR:SLEW 3

Query Syntax

[SOURce:]CURRENT:SLEW?

Returned Parameters

<NR2>

[SOURce:]CURRENT:SLEW[:BOTH]?

This command is used to query the current rising and falling slope (A/uS).

Command Syntax

[SOURce:]CURRENT:SLEW[:BOTH]?

Unit

A /uS

*RST Value

MAXimum

[SOURce:]CURRENT:SLEW:RISE <NRf+>

[SOURce:]CURRENT:SLEW:POSitive <NRf+>

These two commands are equivalent and are used to set the rate of rise of current (A/uS).

Command Syntax

[SOURce:]CURRent:SLEW:RISE <NRf+>

Parameters

<NRf+>

MIN~MAX | MAX | MIN | DEF

Unit

A/uS

*RST Value

1e-2A/us

Examples

CURR:SLEW RISE 3

Query Syntax

[SOURce:]CURRent:SLEW:RISE?

Returned Parameters

<NR2>

[SOURce:]CURRent:SLEW:RISE?

[SOURce:]CURRent:SLEW:POSitive?

These two commands are equivalent and are used to query the rate of rise of current (A/uS).

Command Syntax

[SOURce:]CURRent:SLEW:RISE?

Unit

A/uS

*RST Value

MAXimum

[SOURce:]CURRent:SLEW:FALL <NRf+>

[SOURce:]CURRent:SLEW:NEGative <NRf+>

This command is used to set the current drop rate (A/uS).

Command Syntax

[SOURce:]CURRent:SLEW:FALL <NRf+>

Parameters

MIN~MAX | MAX | MIN | DEF

Unit

A/uS

Examples

CURR:SLEW FALL 3

Query Syntax

[SOURce:]CURRent:SLEW:FALL?

Returned Parameters

<NR2>

[SOURce:]CURRent:SLEW:FALL?

[SOURce:]CURRent:SLEW:NEGative?

These two commands are equivalent and are used to query the current drop rate (A/uS).

Command Syntax

[SOURce:]CURRent:SLEW:FALL?

Parameters

None

Unit

A/uS

*RST Value

MAXimum

[SOURce:]CURRent[:OVER]:PROTection:STATe <BOOL>

Set the status of the overcurrent protection function

Command Syntax

[SOURce:]CURRent[:OVER]:PROTection:STATe <BOOL>

Parameters

0|1|OFF|ON

Examples

CURR:PROT:STAT ON

Query Syntax

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

Returned Parameters

0|1

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

This command is used to query the current protection function status.

Command Syntax

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

*RST Value

MAXimum

[SOURce:]CURRent:PROTection[:LEVel] <NRf+>

This command is used to set the current protection value. If the input current exceeds the software current protection value within the time specified by CURR:PROT:DEL, the input will be closed.

Note: Use CURR:PROT:DEL to prevent current protection caused by instantaneous overcurrent.

Command Syntax

[SOURce:]CURRent:PROTection[:LEVel] <NRf+>

Parameters

0~MAX | MIN | MAX | DEF

Unit

A

Examples

CURR:PROT 3

Query Syntax

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

Returned Parameters

<NR2>

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

This command is used to query the current protection value.

Command Syntax

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

Unit

A

*RST Value

MAXimum

[SOURce:]CURRent[:OVER]:PROTection:DELay <NRf+>

Set the software current protection delay time.

Command Syntax

[SOURce:]CURRent[:OVER]:PROTection:DELay <NRf+>

Parameters

0 to 60s | MIN | MAX | DEF(3s)

*RST Value

3

Examples

CURR:PROT:DEL 20

Query Syntax

[SOURce:]CURRent[:OVER]:PROTection:DELay?

Returned Parameters

<NR2>

[SOURce:]CURRent[:OVER]:PROTection:DELay?

This command is used to query the software current protection delay time.

Command Syntax

[SOURce:]CURRent[:OVER]:PROTection:DELay?

Returned Parameters

<NR2>

[SOURce:]POWer:PROTection[:LEVel] <NRf+>

This command is used to set the power protection value. If the power exceeds the power protection value within the time specified by POW:PROT:DEL, the input is closed.

Note: Use the POW:PROT:DEL command to prevent the instantaneous

power protection situation, which is caused by the edit change of stopping the over-power protection.

Command Syntax

[SOURce:]POWer:PROTection[:LEVel] <NRf+>

Parameters

0 to rated value| MIN | MAX | DEF

Unit

W

Examples

POW:PROT 100

Query Syntax

[SOURce:]POWer:PROTection[:LEVel]?

Returned Parameters

<NR2>

[SOURce:]POWer:PROTection[:LEVel]?

This command is used to query the power protection value.

Command Syntax

[SOURce:]POWer:PROTection[:LEVel]?

Unit

W

*RST Value

MAXimum(high range)

[SOURce:]POWer:PROTection:DELaY <NRf+>

This command is used to set the software power protection delay time.

Command Syntax

[SOURce:]POWer:PROTection:DELaY <NRf+>

Parameters

0 to 60s | MIN | MAX | DEF (3s)

*RST Value

3

Examples

```
POW:PROT:DEL 40
```

Query Syntax

```
[SOURce:]POWer:PROTection:DELay?
```

Returned Parameters

```
<NR2>
```

[SOURce:]POWer:PROTection:DELay?

This command is used to query the software power protection delay time.

Command Syntax

```
[SOURce:]POWer:PROTection:DELay?
```

Returned Parameters

```
<NR2>
```

[SOURce:]POWer:CONFIg[:LEVel] <NRf+>

This command is used to set the hardware power protection value (corresponding to P-LIMIT in each mode).

Command Syntax

```
[SOURce:]POWer:CONFIg[:LEVel] <NRf+>
```

Parameters

```
0 to rated value| MIN | MAX | DEF
```

Examples

```
POW:CONF 40
```

Query Syntax

```
[SOURce:]POWer:CONFIg[:LEVel]?
```

Returned Parameters

```
<NR2>
```

[SOURce:]POWer:CONFIg[:LEVel]?

This command is used to query the hardware power protection value (corresponding to P-LIMIT in each mode).

Command Syntax

```
[SOURce:]POWer:CONFIg[:LEVel]?
```

Returned Parameters

<NR2>

[SOURce:]VOLTage:[LEVel:]ON <NRf+>

This command is used to set the starting voltage value of the load.

Command Syntax

[SOURce:]VOLTage[:LEVel]:ON <NRf+>

Parameters

MIN~MAX | MIN | MAX | DEF

Unit

V

***RST Value**

0

Examples

VOLT:ON 3

Query Syntax

[SOURce:]VOLTage[:LEVel]:ON?

Returned Parameters

<NR2>

[SOURce:]VOLTage:[LEVel:]ON?

This command is used to query the starting voltage value of the load.

Command Syntax

[SOURce:]VOLTage[:LEVel]:ON?

Unit

V

***RST Value**

1

[SOURce:]CURRent:TRANSient:MODE <CRD>**[SOURce:]DYNamic:MODE <CRD>**

These two commands are equivalent and are used to set the trigger mode of the transient generator in CC mode

Command Syntax

[SOURce:]DYNamic:MODE <CRD>

Parameters

CONTInuous | PULSe | TOGGle

*RST Value

Not involved

Examples

DYN:MODE PULSe

Query Syntax

[SOURce:]DYNamic:MODE?

Returned Parameters

CONTInuous | PULSe | TOGGle

[SOURce:]CURRENT:TRANSient:MODE?

[SOURce:]DYNamic:MODE?

These two commands are equivalent and are used to query the trigger mode of the transient generator in CC mode.

Command Syntax

[SOURce:]DYNamic:MODE?

Returned Parameters

CONTInuous | PULSe | TOGGle

[SOURce:]VOLTage:[LEVel:]OFF <NRf+>

This command is used to set the start unloading voltage value of the load.

Command Syntax

[SOURce:]VOLTage[:LEVel]:OFF <NRf+>

Parameters

0 to rated value | MIN | MAX | DEF

Unit

V

*RST Value

0

Examples

VOLT:OFF 2

Query Syntax

[SOURce:]VOLTage[:LEVel]:OFF?

Returned Parameters

<NR2>

[SOURce:]VOLTage:[LEVel:]OFF?

This command is used to query the start unloading voltage value of the load.

Command Syntax

[SOURce:]VOLTage[:LEVel]:OFF?

Unit

V

*RST Value

0

[SOURce:]VOLTage:LATCh[:STATe] <BOOL>

This command is used to set the type of VON voltage unloading.

Command Syntax

[SOURce:]VOLTage:LATCh[:STATe] <BOOL>

Parameters

0|1|OFF|ON

*RST Value

Not involved

Examples

VOLT:LATC OFF

Query Syntax

[SOURce:]VOLTage:LATCh[:STATe]?

Returned Parameters

0|1

[SOURce:]VOLTage:LATCh[:STATe]?

This command is used to query the type of VON voltage unloading.

Command Syntax

[SOURce:]VOLTage:LATCh[:STATe]?

Returned Parameters

0|1

[SOURce:]VOLTage:ILIMIT <NRf+>

This command is used to set the current limit value in CV mode.

Command Syntax

[SOURce:]VOLTage:ILIMIT <NRf+>

Parameters

0~rated value | MIN | MAX | DEF

Unit

A

Examples

VOLT:ILIMIT 13

Query Syntax

[SOURce:]VOLTage:ILIMIT?

Returned Parameters

<NR2>

[SOURce:]VOLTage:ILIMIT?

This command is used to query the current limit value in CV mode.

Command Syntax

[SOURce:]VOLTage:ILIMIT?

Unit

A

Returned Parameters

<NR2>

[SOURce:]VOLTage:SLOPe[:RANGe] <CRD>

This command is used to set the voltage change rate mode in CV mode.

Command Syntax

[SOURce:]VOLTage:SLOPe[:RANGe] <CRD>

Parameters

HIGH | LOW

*RST Value

High

Examples

VOLT:SLOP HIGH

Query Syntax

[SOURce:]VOLTage:SLOPe[:RANGe]?

Returned Parameters

HIGH | LOW

[SOURce:]VOLTage:SLOPe[:RANGe]?

This command is used to query the voltage change rate mode in CV mode.

Command Syntax

[SOURce:]VOLTage:SLOPe[:RANGe]?

Returned Parameters

HIGH | LOW

[SOURce:]VOLTage:TRANSient:MODE <CRD>

This command is used to set the trigger mode of the transient generator in CV mode.

Command Syntax

[SOURce:]VOLTage:TRANSient:MODE <CRD>

Parameters

CONTInuous | PULSe | TOGGle

*RST Value

Not involved

Examples

VOLT:TRAN:MODE PULSe

Query Syntax

[SOURce:]VOLTage:TRANSient:MODE?

Returned Parameters

CONTInuous | PULSe | TOGGle

[SOURce:]VOLTage:TRANSient:MODE?

This command is used to query the trigger mode of the transient generator in CV mode.

Command Syntax

[SOURce:]VOLTage:TRANSient:MODE?

Returned Parameters

CONTInuous | PULSe | TOGGle

[SOURce:]VOLTage:TRANSient:ALEVel <NRf+>

This command is used to set the A state voltage value of the transient generator CV mode

Command Syntax

[SOURce:]VOLTage:TRANSient:ALEVel <NRf+>

Parameters

0.1 to Range | MIN | MAX | DEF (Range)

*RST Value

Range

Examples

VOLT:TRAN:ALEV 0.7

Query Syntax

[SOURce:]VOLTage:TRANSient:ALEVel?

Returned Parameters

<NR2>

[SOURce:]VOLTage:TRANSient:ALEVel?

This command is used to query the A state voltage value of the transient generator CV mode

Command Syntax

[SOURce:]VOLTage:TRANSient:ALEVel?

Returned Parameters

<NR2>

[SOURce:]VOLTage:TRANSient:BLEVel <NRf+>

This command is used to set the B state voltage value of the transient generator CV mode

Command Syntax

[SOURce:]VOLTage:TRANSient:BLEVel <NRf+>

Parameters

0.1 to Range | MIN | MAX | DEF (Range)

*RST Value

Range

Examples

VOLT:TRAN:BLEV 0.7

Query Syntax

[SOURce:]VOLTage:TRANSient:BLEVel?

Returned Parameters

<NR2>

[SOURce:]VOLTage:TRANSient:BLEVel?

This command is used to query the B state voltage value of the transient generator CV mode

Command Syntax

[SOURce:]VOLTage:TRANSient:BLEVel?

Returned Parameters

<NR2>

[SOURce:]VOLTage:TRANSient:AWIDTH <NRf+>

This command is used to set the switching pulse width of the input voltage

Command Syntax

[SOURce:]VOLTage:TRANSient:AWIDTH <NRf+>

Parameters

100.0uS to 3600.0S | MIN | MAX | DEF(1e-3s)

*RST Value

1e-3s

Examples

VOLT:TRAN:AWID 500

Query Syntax

[SOURce:]VOLTage:TRANSient:AWIDTH?

Returned Parameters

<NR2>

[SOURce:]VOLTage:TRANSient:AWIDth?

This command is used to query the switching pulse width of the input voltage

Command Syntax

[SOURce:]VOLTage:TRANSient:AWIDth?

Returned Parameters

<NR2>

[SOURce:]VOLTage:TRANSient:BWIDth <NRf+>

This command is used to set the switching pulse width of the input voltage

Command Syntax

[SOURce:]VOLTage:TRANSient:BWIDth <NRf+>

Parameters

100.0uS to 3600.0S | MIN | MAX | DEF(1e-3s)

*RST Value

1e-3s

Examples

VOLT:TRAN:BWID 500

Query Syntax

[SOURce:]VOLTage:TRANSient:BWIDth?

Returned Parameters

<NR2>

[SOURce:]VOLTage:TRANSient:BWIDth?

This command is used to query the switching pulse width of the input voltage.

Command Syntax

[SOURce:]VOLTage:TRANSient:BWIDth?

Returned Parameters

<NR2>

[SOURce:]CURRent[:LEVel][:IMMediate][:AMPLitude] <NRf+>

This command is used to set the constant current in CC mode.

Command Syntax

[SOURce:]CURRent[:LEVel][:IMMediate][:AMPLitude] <NRf+>

Parameters

0 to Range | MIN| MAX | DEF (0A)

Unit

0A

*RST Value

MINimum

Examples

CURR 5

Query Syntax

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

Returned Parameters

<NR2>

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

This command is used to query the constant current in CC mode.

Command Syntax

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

Parameters

None

Unit

A

*RST Value

MINimum

[SOURce:]VOLTage[:LEVel][:IMMediate][:AMPLitude] <NRf+>

This command is used to set the constant voltage in CV mode.

Command Syntax

[SOURce:]VOLTage[:LEVel][:IMMediate][:AMPLitude] <NRf+>

Parameters

0.1 to Range | MIN | MAX | DEF (Range)

Unit

V

*RST Value

Range

Examples

VOLT 5

Query Syntax

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

Returned Parameters

<NR2>

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

This command is used to query the constant voltage in CV mode.

Command Syntax

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

Parameters

None

Unit

V

*RST Value

MAXimum

[SOURce:]POWER[:LEVel][:IMMediate][:AMPLitude] <NRf+>

This command is used to set the constant power in CW mode.

Command Syntax

[SOURce:]POWER[:LEVel][:IMMediate][:AMPLitude] <NRf+>

Parameters

0 to rated value | MIN | MAX | DEF

Unit

W

Examples

POW 10

Query Syntax

[SOURce:]POWer[:LEVel][:IMMediate][:AMPLitude]?

Returned Parameters

<NR2>

[SOURce:]POWer[:LEVel][:IMMediate][:AMPLitude]?

This command is used to query the constant power in CW mode.

Command Syntax

[SOURce:]POWer[:LEVel][:IMMediate][:AMPLitude]?

Parameters

None

Unit

W

[SOURce:]POWer:ILIMIT <NRf+>

This command is used to set the current limit value in CW mode

Command Syntax

[SOURce:]POWer:ILIMIT <NRf+>

Parameters

0 to rated value | MIN | MAX | DEF

Unit

A

Examples

POW ILIMIT 10

Query Syntax

[SOURce:]POWer:ILIMIT?

Returned Parameters

<NR2>

[SOURce:]POWer:ILIMIT?

This command is used to query the current limit value in CW mode

Command Syntax

[SOURce:]POWer:ILIMIT?

Unit

A

Returned Parameters

<NR2>

[SOURce:]POWer:TRANsient:MODE <CRD>

This command is used to set the trigger mode of the transient generator in CW mode

Command Syntax

[SOURce:]POWer:TRANsient:MODE <CRD>

Parameters

CONTInuous | PULSe | TOGGle

*RST Value

Not involved

Examples

POW:TRAN:MODE PULSe

Query Syntax

[SOURce:]POWer:TRANsient:MODE?

Returned Parameters

CONTInuous | PULSe | TOGGle

[SOURce:]POWer:TRANsient:MODE?

This command is used to query the trigger mode of the transient generator in CW mode

Command Syntax

[SOURce:]POWer:TRANsient:MODE?

Returned Parameters

CONTInuous | PULSe | TOGGle

[SOURce:]POWer:TRANsient:ALEVel <NRf+>

This command is used to set the power value of the A state in the CW mode of the transient generator.

Command Syntax

```
[SOURce:]POWer:TRANsient:ALEVel <NRf+>
```

Parameters

0 to rated value | MIN | MAX | DEF

Examples

```
POW:TRAN:ALEV 34
```

Query Syntax

```
[SOURce:]POWer:TRANsient:ALEVel?
```

Returned Parameters

< NR2 >

[SOURce:]POWer:TRANsient:ALEVel?

This command is used to query the power value of the A state in the CW mode of the transient generator.

Command Syntax

```
[SOURce:]POWer:TRANsient:ALEVel?
```

Returned Parameters

<NR2>

[SOURce:]POWer:TRANsient:BLEVel <NRf+>

This command is used to set the power value of the B state in the CW mode of the transient generator.

Command Syntax

```
[SOURce:]POWer:TRANsient:BLEVel <NRf+>
```

Parameters

0 to rated value | MIN | MAX | DEF

Examples

```
POW:TRAN:BLEV
```

Query Syntax

```
[SOURce:]POWer:TRANsient:BLEVel?
```

Returned Parameters

< NR2 >

[SOURce:]POWer:TRANsient:BLEVel?

This command is used to query the power value of the B state in the CW mode of the transient generator.

Command Syntax

[SOURce:]POWer:TRANsient:BLEVel?

Return Parameters

< NR2 >

[SOURce:]POWer:TRANsient:AWIDth <NRf+>

This command is used to set the switching pulse width of the input power.

Command Syntax

[SOURce:]POWer:TRANsient:AWIDth <NRf+>

Parameters

100.0uS to 3600.0S | MIN | MAX | DEF (1e-3s)

*RST Value

1e-3s

Examples

POW:TRAN:AWID 100

Query Syntax

[SOURce:]POWer:TRANsient:AWIDth?

Returned Parameters

<NR2 >

[SOURce:]POWer:TRANsient:AWIDth?

This command is used to query the switching pulse width of the input power.

Command Syntax

[SOURce:]POWer:TRANsient:AWIDth?

Returned Parameters

<NR2 >

[SOURce:]POWer:TRANsient:BWIDth <NRf+>

This command is used to set the switching pulse width of the input power.

Command Syntax

[SOURce:]POWer:TRANsient:BWIDth <NRf+>

Parameters

100.0uS to 3600.0S | MIN | MAX | DEF (1e-3s)

*RST Value

1e-3s

Examples

POW:TRAN:BWID 100

Query Syntax

[SOURce:]POWer:TRANsient:BWIDth?

Returned Parameters

<NR2 >

[SOURce:]POWer:TRANsient:BWIDth?

This command is used to query the switching pulse width of the input power.

Command Syntax

[SOURce:]POWer:TRANsient:BWIDth?

Returned Parameters

<NR2 >

[SOURce:]RESistance[:LEVel][:IMMediate][:AMPLitude] <NRf+>

This command is used to set the constant resistance in CR mode.

Command Syntax

[SOURce:]RESistance[:LEVel][:IMMediate][:AMPLitude] <NRf+>

Parameters

MIN~MAX| MIN | MAX | DEF

Unit

ohm

Examples

RES 5

Query Syntax

[SOURce:]RESistance[:LEVel][:IMMediate][:AMPLitude]?

Returned Parameters

<NR2>

[SOURce:]RESistance[:LEVel][:IMMediate][:AMPLitude]?

This command is used to query the constant resistance in CR mode.

Command Syntax

[SOURce:]RESistance[:LEVel][:IMMediate][:AMPLitude]?

Parameters

None

Unit

ohm

*RST Value

MAXimum

[SOURce:]RESistance:ILIMIT <NRf+>

This command is used to set the current limit value in CR mode.

Command Syntax

[SOURce:]RESistance:ILIMIT <NRf+>

Parameters

0 to rated value | MIN | MAX | DEF

Unit

A

Examples

RES:ILIMIT 5

Query Syntax

[SOURce:]RESistance:ILIMIT?

Returned Parameters

<NR2>

[SOURce:]RESistance:ILIMIT?

This command is used to query the current limit value in CR mode.

Command Syntax

[SOURce:]RESistance:ILIMIT?

Returned Parameters

<NR2>

[SOURce:]RESistance:TRANSient:MODE <CRD>

This command is used to set the trigger mode of the transient generator in CR mode.

Command Syntax

[SOURce:]RESistance:TRANSient:MODE <CRD>

Parameters

CONTInuous|PULSe|TOGGle

Examples

RES:TRAN:MODE PULSe

Query Syntax

[SOURce:]RESistance:TRANSient:MODE?

Returned Parameters

CONTInuous|PULSe|TOGGle

[SOURce:]RESistance:TRANSient:MODE?

This command is used to query the trigger mode of the transient generator in CR mode.

Command Syntax

[SOURce:]RESistance:TRANSient:MODE?

Returned Parameters

CONTInuous|PULSe|TOGGle

[SOURce:]RESistance:TRANSient:ALEVel <NRf+>

This command is used to set the resistance value of the A state in the CR mode of the transient generator.

Command Syntax

[SOURce:]RESistance:TRANSient:ALEVel <NRf+>

Parameters

MIN~MAX | MIN | MAX | DEF

Examples

```
RES:TRAN:ALEV 600
```

Query Syntax

```
[SOURce:]RESistance:TRANSient:ALEVel?
```

Returned Parameters

```
<NR2>
```

[SOURce:]RESistance:TRANSient:ALEVel?

This command is used to query the resistance value of the A state in the CR mode of the transient generator.

Command Syntax

```
[SOURce:]RESistance:TRANSient:ALEVel?
```

Returned Parameters

```
<NR2>
```

[SOURce:]RESistance:TRANSient:BLEVel <NRf+>

This command is used to set the resistance value of the B state in the CR mode of the transient generator.

Command Syntax

```
[SOURce:]RESistance:TRANSient:BLEVel <NRf+>
```

Parameters

```
MIN~MAX | MIN | MAX | DEF
```

Examples

```
RES:TRAN:ALEV 60
```

Query Syntax

```
[SOURce:]RESistance:TRANSient:BLEVel?
```

Returned Parameters

```
<NR2>
```

[SOURce:]RESistance:TRANSient:BLEVel?

This command is used to query the resistance value of the B state in the CR mode of the transient generator.

Command Syntax

```
[SOURce:]RESistance:TRANSient:BLEVel?
```

Returned Parameters

<NR2>

[SOURce:]RESistance:TRANSient:AWIDth <NRf+>

This command is used to set the switching pulse width of the input resistance.

Command Syntax

`[SOURce:]RESistance:TRANSient:AWIDth <NRf+>`

Parameters

100.0uS to 3600.0S | MIN | MAX | DEF (1e-3s)

Examples

RES:TRAN:AWID 600

Query Syntax

`[SOURce:]RESistance:TRANSient:AWIDth?`

Returned Parameters

<NR2>

[SOURce:]RESistance:TRANSient:AWIDth?

This command is used to query the switching pulse width of the input resistance.

Command Syntax

`[SOURce:]RESistance:TRANSient:AWIDth?`

Returned Parameters

<NR2>

[SOURce:]RESistance:LED[:STATe] <BOOL>

Enable or disable load CR mode LED test.

Command Syntax

`[SOURce:]RESistance:LED[:STATe] <BOOL+>`

Parameters

0 | 1 | OFF | ON

Examples

RES:LED ON

Query Syntax

`[SOURce:]RESistance:LED[:STATe]?`

Returned Parameters

0 | 1

[SOURCE:]RESistance:LED[:STATE]?

Query the enable or disable status of the LED test in load CR mode.

Command Syntax

[SOURCE:]RESistance:LED[:STATE]?

Returned Parameters

0 | 1

[SOURCE:]RESistance:TRANSient:BWIDth <NRf+>

This command is used to set the switching pulse width of the input resistance.

Command Syntax

[SOURCE:]RESistance:TRANSient:BWIDth <NRf+>

Parameters

100.0uS to 3600.0S | MIN | MAX | DEF (1e-3s)

Examples

RES:TRAN:BWID 600

Query Syntax

[SOURCE:]RESistance:TRANSient:BWIDth?

Returned Parameters

<NR2>

[SOURCE:]RESistance:TRANSient:BWIDth?

This command is used to query the switching pulse width of the input resistance.

Command Syntax

[SOURCE:]RESistance:TRANSient:BLEVel?

Returned Parameters

<NR2>

[SOURCE:]DYNamic:HIGH[:LEVel] <NRf+>

[SOURCE:]CURRent:TRANSient:ALEVel <NRf+>

These two commands are equivalent and are used to set the high-level current

in dynamic mode.

Command Syntax

[SOURce:]DYNamic:HIGH[:LEVel] <NRf+>

Parameters

0 to Range | MIN | MAX | DEF (0A)

Unit

A

*RST Value

0

Examples

DYN:HIGH 10

Query Syntax

[SOURce:]DYNamic:HIGH[:LEVel]?

Returned Parameters

<NR2>

[SOURce:]DYNamic:HIGH[:LEVel]?

[SOURce:]CURRent:TRANsient:ALEVel?

These two commands are equivalent and are used to query the high-level current in dynamic mode.

Command Syntax

[SOURce:]DYNamic:HIGH[:LEVel]?

Parameters

None

Unit

A

*RST Value

0

Examples

DYN:HIGH?

[SOURce:]DYNamic:HIGH:DWELI <NRf+>

[SOURce:]CURRent:TRANSient:AWIDth <NRf+>

These two commands are equivalent and are used to set the duration of high-level load current in dynamic mode.

Command Syntax

[SOURce:]DYNamic:HIGH:DWELI <NRf+>

Parameters

20.0uS to 3600.0S | MIN | MAX | DEF (1e-3s)

Unit

S

*RST Value

1e-3s

Examples

DYN:HIGH:DWELI 0.01

Query Syntax

[SOURce:]DYNamic:HIGH:DWELI?

Returned Parameters

<NR2>

[SOURce:]DYNamic:HIGH:DWELI?

[SOURce:]CURRent:TRANSient:AWIDth?

This command is used to query the duration of high-level load current in dynamic mode.

Command Syntax

[SOURce:]DYNamic:HIGH:DWELI?

Parameters

None

Unit

S

*RST Value

MIN

[SOURce:]DYNamic:LOW[:LEVel] <NRf+>

[SOURce:]CURRent:TRANsient:BLEVel <NRf+>

These two commands are equivalent and are used to set the low-level current in dynamic mode.

Command Syntax

[SOURce:]DYNamic:LOW[:LEVel] <NRf+>

Parameters

0 to Range | MIN | MAX | DEF (Range)

Unit

A

*RST Value

Range

Examples

DYN:LOW 1

Query Syntax

[SOURce:]DYNamic:LOW[:LEVel]?

Returned Parameters

<NR2>

[SOURce:]DYNamic:LOW[:LEVel]?

[SOURce:]CURRent:TRANsient:BLEVel?

These two commands are equivalent and are used to query the low-level load current in dynamic mode.

Command Syntax

[SOURce:]DYNamic:LOW[:LEVel]?

Parameters

None

Unit

A

*RST Value

0

[SOURce:]DYNamic:LOW:DWELI <NRf+>

[SOURce:]CURRent:TRANSient:BWIDth <NRf+>

These two commands are equivalent and are used to set the duration of the low-position current in the dynamic mode.

Command Syntax

[SOURce:]DYNamic:LOW:DWELI <NRf+>

Parameters

20.0uS to 3600.0S | MIN | MAX | DEF (1e-3s)

Unit

S

*RST Value

1e-3s

Examples

DYN:LOW:DWEL 1

Query Syntax

[SOURce:]DYNamic:LOW:DWELI?

Returned Parameters

<NR2>

[SOURce:]DYNamic:LOW:DWELI?

[SOURce:]CURRent:TRANSient:BWIDth?

This command is used to query the duration of low-level load current in dynamic mode.

Command Syntax

[SOURce:]DYNamic:LOW:DWELI?

Parameters

None

Unit

S

*RST Value

0.001

Chapter 7 Measure Commands

This signal measurement command is used to obtain readings. You can use this command to control the measurement process.

Note: The fetch instruction and the measure instruction can also read various parameter values. The difference is that fetch reads the last measured value, while measure is the re-measured value. The fetch is faster in speed, but the accuracy measure is higher.

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

FETCh[:SCALar]:VOLTage[:DC]?

This command is used to read the average value of the input voltage of the instrument.

Command Syntax

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

Parameters

None

Examples

MEAS:VOLT?

Returned Parameters

<NR2>

MEASure[:SCALar]:VOLTage:MAXimum?

FETCh[:SCALar]:VOLTage[:AMPLitude]:MAXimum[:INSTant]?

This command is used to read the maximum input voltage of the instrument.

Command Syntax

MEASure[:SCALar]:VOLTage:MAXimum?

Parameters

None

Examples

MEAS:VOLT:MAX?

Returned Parameters

<NR2>

MEASure[:SCALar]:VOLTage:MINimum?

FETCh[:SCALar]:VOLTage[:AMPLitude]:MINimum[:INSTant]?

This command is used to read the minimum input voltage of the instrument.

Command Syntax

MEASure[:SCALar]:VOLTage:MINimum?

Parameters

None

Examples

MEAS:VOLT:MIN?

Returned Parameters

<NR2>

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

FETCh[:SCALar]:CURRent[:DC]?

This command is used to query the average current value.

Command Syntax

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

Parameters

None

Examples

MEAS:CURR?

Returned Parameters

<NR2>

MEASure[:SCALar]:CURRent:MAXimum?

FETCh[:SCALar]:CURRent[:AMPLitude]:MAXimum[:INSTant]?

This command is used to query the current peak value.

Command Syntax

MEASure[:SCALar]:CURRent:MAXimum?

Parameters

None

Examples

MEAS:CURR:MAX?

Returned Parameters

<NR2>

MEASure[:SCALar]:CURRent:MINimum?

FETCh[:SCALar]:CURRent[:AMPLitude]:MINimum[:INSTant]?

This command is used to query the minimum current.

Command Syntax

MEASure[:SCALar]:CURRent:MINimum?

Parameters

None

Examples

MEAS:CURR:MIN?

Returned Parameters

<NR2>

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

FETCh[:SCALar]:POWer[:DC]?

This command is used to query the average power value.

Command Syntax

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

Parameters

None

Examples

MEAS:POWer?

Returned Parameters

<NR2>

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

FETCh[:SCALar]:RESistance[:DC]?

This command is used to query the equivalent impedance.

Command Syntax

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

Parameters

None

Examples

MEAS:RES?

Returned Parameters

<NR2>

MEASure:CAPacity?

This command is used to read the discharged battery capacity (clear to zero).

Command Syntax

MEASure:CAPacity?

Parameters

None

Examples

MEASure:CAPacity?

Returned Parameters

<NR2>

FETCh:CAPacity?

This command is used to read the discharged battery capacity (not cleared).

Command Syntax

FETCh:CAPacity?

Parameters

None

Examples

FETCh:CAPacity?

Returned Parameters

<NR2>

MEASure:TIME?

FETCh:TIME?

This command is used to read the integration time of the instrument.

Command Syntax

MEASure:TIME?

Parameters

None

Examples

MEASure:TIME?

Returned Parameters

<NR2>

MEASure:TEMPerature?

FETCh:TEMPerature?

This command is used to get the internal temperature of the chassis.

Command Syntax

MEASure:TEMPerature?

Parameters

None

Examples

MEASure:TEMPerature?

Returned Parameters

<NR2>

MEASure:VOLTage:RIPPlE?

MEASure[:SCALar]:VOLTage:PTPeak?

This command is used to read the peak-to-peak voltage.

Command Syntax

MEASure:VOLTage:RIPPlE?

Parameters

None

Examples

MEASure:VOLTage:RIPPlE?

Returned Parameters

<NR2>

MEASure:CURRent:RIPPlE?

MEASure[:SCALar]:CURRent:PTPeak?

This command is used to read the current peak-to-peak value.

Command Syntax

MEASure:CURRent:RIPPlE?

Parameters

None

Examples

MEASure:CURRent:RIPPlE?

Returned Parameters

<NR2>

Chapter 8 TRIGger Commands

TRIGger[:IMMediate]

This command is used to execute a trigger.

Command Syntax

TRIGger[:IMMediate]

Parameters

None

Examples

TRIG

TRIGger:SOURce <CPD>

Used to set the trigger mode.

MANual: Triggered by the Trigger button on the panel

BUS: Command trigger (*TRG or TRIGger[:IMMediate])

EXTernal: external signal trigger (trigger terminal on the rear panel)

Command Syntax

TRIGger:SOURce <CPD>

Parameters

<CPD>

MANual|BUS|EXTernal

Returned Parameters

Not involved

Examples

TRIGger:SOURce EXT

TRIGger:SOURce?

Used to query the trigger mode.

Command Syntax

TRIGger:SOURce?

Parameters

None

Returned Parameters

<CRD>

Chapter 9 SENSE Commands

SENSE:AVERage <NRf+>

This command is used to set the exponential power value of the filter parameter.

Command Syntax

```
SENSE:AVERage <NRf+>
```

Parameters

7 to 14 | MIN | MAX | DEF

Examples

```
SENSE:AVERage 10
```

SENSE:AVERage?

This command is used to query the exponential power value of the filter parameter.

Command Syntax

```
SENSE:AVERage?
```

Parameters

None

Returned Parameters

```
<NRf+>
```

SENSE:TIME:VOLTage1 <NRf+>

This command is used to set the voltage point 1 when the load measures the rise and fall time of the output voltage of the DUT.

Command Syntax

```
SENSE:TIME:VOLTage1 <NRf+>
```

Parameters

0 to rated value | MIN | MAX | DEF

Examples

```
SENSE:TIME:VOLTage1 3
```

SENSE:TIME:VOLTage1?

This command is used to query the voltage point 1 when the load measures the rise and fall time of the output voltage of the DUT.

Command Syntax

SENSe:TIME:VOLTage1?

Parameters

None

Returned Parameters

<NRf+>

SENSe:TIME:VOLTage2 <NRf+>

This command is used to set the voltage point 2 when the load measures the rise and fall time of the output voltage of the DUT.

Command Syntax

SENSe:TIME:VOLTage2 <NRf+>

Parameters

0 to rated value| MIN | MAX | DEF

Examples

SENSe:TIME:VOLTage2 3

SENSe:TIME:VOLTage2?

This command is used to query the voltage point 2 when the load measures the rise and fall time of the output voltage of the DUT.

Command Syntax

SENSe:TIME:VOLTage2?

Parameters

None

Returned Parameters

<NRf+>

SENSe[:REMOte][:STATe] <bool>

This command is used to set the remote measurement state of the load.

Command Syntax

SENSe[:REMOte][:STATe] <bool>

Parameters

0|1|OFF|ON

Examples

SENS ON

SENSe[:REMOte][:STATe]?

This command is used to query the remote measurement state of the load.

Command Syntax

SENSe[:REMOte][:STATe]?

Parameters

0|1|OFF|ON

SENSe:TIME:VOLTage:UP?

This command is used to read the voltage rise time.

Command Syntax

SENSe:TIME:VOLTage:UP?

Parameters

None

Returned Parameters

<NRf+>

SENSe:TIME:VOLTage:DOWN?

This command is used to read the voltage fall time.

Command Syntax

SENSe:TIME:VOLTage:DOWN?

Parameters

None

Returned Parameters

<NRf+>

SENSe:VOLTage:POSitive:PULSe?

This command is used to read the voltage positive pulse time.

Command Syntax

SENSe:VOLTage:POSitive:PULSe?

Parameters

None

Returned Parameters

<NRf+>

SENSe:VOLTage:NEGative:PULSe?

This command is used to read the voltage reverse pulse time.

Command Syntax

SENSe:VOLTage:NEGative:PULSe?

Parameters

None

Returned Parameters

<NRf+>

Chapter 10 List Commands

[SOURce:]LIST:RANGe <NRf+>

Set the current range in LIST mode.

Command Syntax

[SOURce:]LIST:RANGe <NRf+>

Parameters

0~MAX | MAX | MIN

Examples

LIST:RANGe 22

Query Syntax

[SOURce:]LIST:RANGe?

Return Parameters

<NR2>

[SOURce:]LIST:RANGe?

Query the current range in LIST mode.

Command Syntax

[SOURce:]LIST:RANGe?

Returned Parameters

<NR2>

[SOURce:]LIST:COUNT < NRf+>

This command is used to set the cycle times of LIST.

Command Syntax

[SOURce:]LIST:COUNT <NRf+>

Parameters

1 to 65535 | MIN | MAX

Examples

LIST:COUN 5000

Query Syntax

[SOURce:]LIST:COUNT?

Return Parameters

<NR2>

[SOURCE:]LIST:COUNT?

This command is used to query the cycle times of LIST.

Command Syntax

[SOURCE:]LIST:COUNT?

Return Parameters

<NR2>

[SOURCE:]LIST:STEP < NRf+>

This command is used to set the total step number of the list.

Command Syntax

[SOURCE:]LIST:STEP <NRf+>

Parameters

2 to 84 | MIN | MAX

Examples

LIST:STEP 30

Query Syntax

[SOURCE:]LIST:STEP?

Return Parameters

<NR2>

[SOURCE:]LIST:STEP?

This command is used to query the total step number of the list.

Command Syntax

LIST:STEP?

Return Parameters

<NR2>

[SOURCE:]LIST:SLEW[:BOTH] <NR1>,<NRf+>

This command is used to set the current slope of each step.

Command Syntax

[SOURCE:]LIST:SLEW[:BOTH] <NR1>,<NRf+>

Parameters

<NR1>,<NRf+>
1 to step, MIN~MAX|MIN|MAX

Examples

LIST:SLEW 1, 0.4

Query Syntax

[SOURce:]LIST:SLEW[:BOTH]? <NR1>

Return Parameters

<NR2>

[SOURce:]LIST:SLEW[:BOTH]? <NR1>

This command is used to query the current slope of each step.

Command Syntax

[SOURce:]LIST:SLEW[:BOTH]? <NR1>

Return Parameters

<NR2>

[SOURce:]LIST:LEVel <NR1>,<NRf+>

This command is used to set the current value of each step.

Command Syntax

[SOURce:]LIST:LEVel <NRf+>

Parameters

<NR1> , <NRf+>
1 to step, 0 to range| MIN | MAX

Examples

LIST:LEV 1, 0.4

Query Syntax

[SOURce:]LIST:LEVel? <NR1>

Return Parameters

<NR2>

[SOURce:]LIST:LEVel? <NR1>

This command is used to query the current value of each step.

Command Syntax

[SOURce:]LIST:LEVel? <NR1>

Return Parameters

<NR2>

[SOURce:]LIST:WIDTh <NR1>,<NRf+>

This command is used to set the pulse width time of each step of the List.

Command Syntax

[SOURce:]LIST:WIDTh <NR1>,<NRf+>

Parameters

<NR1> , <NRf+>

1 to step, 20uS to 3600s|MAX|MIN

Examples

LIST:WIDT 1, 160

Query Syntax

[SOURce:]LIST:WIDT?

Return Parameters

<NR2>

[SOURce:]LIST:WIDT?

This command is used to query the pulse width time of each step of the List.

Command Syntax

[SOURce:]LIST:WIDT?

Return Parameters

<NR2>

[SOURce:]LIST:SAV < NR1>

This command is used to store the list file in the memory.

Command Syntax

[SOURce:]LIST:SAV <NR1>

Parameters

<NR1>

1 to 7

Examples

LIST:SAV 3

[SOURCE:]LIST:RCL < NR1 >

This command is used to recall the list file stored in the memory.

Command Syntax

[SOURCE:]LIST:RCL <NR1>

Parameters

<NR1>

1 to 7

Examples

LIST:RCL

[SOURCE:]LIST:RUN:STEP?

This command is used to query the running step number of LIST.

Command Syntax

[SOURCE:]LIST:RUN:STEP?

[SOURCE:]LIST:RUN:REPeat?

This command is used to query the number of times the LIST is currently running repeatedly.

Chapter 11 QC Commands

QC2QC INIT

This command is used to set the quick charge mode to QC2.0 mode.

Command Syntax

QC2QC INIT

QC2QC:classA <CPD>

This command is used to set the voltage value in QC2.0 mode.

Command Syntax

QC2QC:classA <CPD>

Parameters

<5|9|12>

Examples

QC2QC:classA 9

Query Syntax

QC2QC:classA?

Return Parameters

<BOOL>

QC2QC:classA?

This command is used to query whether the voltage regulation is ready (BOOL)

Command Syntax

QC2QC:classA?

Return Parameters

<BOOL>

QC2QC:classB <CPD>

This command is used to set the voltage value in QC2.0 mode.

Command Syntax

QC2QC:classB <CPD>

Parameters

<5|9|12|20>

Examples

QC2QC:classB 9

Query Syntax

QC2QC:classB?

Return Parameters

<BOOL>

QC2QC:classB?

This command is used to query whether the voltage regulation is ready (BOOL)

Command Syntax

QC2QC:classB?

Return Parameters

<BOOL>

QC3QC INIT

This command is used to set the quick charge mode to QC3.0 mode.

Command Syntax

QC3QC INIT

QC3QC:classA <NRf+>

This command is used to set the voltage value in QC3.0 mode.

Command Syntax

QC3QC:classA < NRf+>

Parameters

3.6V-12V|MAX|MIN

Examples

QC3QC:classA 9

Query Syntax

QC3QC:classA?

Return Parameters

<NR2>

QC3QC:classA?

This command is used to query whether the voltage regulation is ready (BOOL)

Command Syntax

QC3QC:classA?

Return Parameters

<NR2>

QC3QC:classB <NRf+>

This command is used to set the voltage value in QC3.0 mode.

Command Syntax

QC3QC:classB <NRf+>

Parameters

3.6V-20V|MAX|MIN

Examples

QC3QC:classB 9

Query Syntax

QC3QC:classB?

Return Parameters

<NR2>

QC3QC:classB?

This command is used to query whether the voltage regulation is ready (BOOL)

Command Syntax

QC3QC:classB?

Return Parameters

<NR2>

PE1PE INIT

This command is used to set the quick charge mode to PE1.0 mode.

PE1PE:VOLT <CPD>

This command is used to set the voltage value in PE1.0 mode.

Command Syntax

PE1PE:VOLT <CPD>

Parameters

<5|7|9|12>

Examples

PE1PE:VOLT 12

Query Syntax

PE1PE:VOLT?

Return Parameters

<BOOL>

PE1PE:VOLT?

Query whether the voltage regulation is ready

Command Syntax

PE1PE:VOLT?

Return Parameters

<BOOL>

PE2PE INIT

This command is used to set the quick charge mode to PE2.0 mode.

Command Syntax

PE2PE INIT

PE2PE:VOLT <NRf+>

This command is used to set the voltage value in PE2.0 mode

Command Syntax

PE2PE:VOLT <NRf+>

Parameters

5V-20V|MAX|MIN

Examples

PE2PE:VOLT 12

Query Syntax

PE2PE:VOLT?

Return Parameters

<BOOL>

PE2PE:VOLT?

This command is used to query whether the voltage regulation is ready

Command Syntax

PE2PE:VOLT?

Return Parameters

<BOOL>

PD2PD INIT

This command is used to set the quick charge mode to PD2.0 mode.

Command Syntax

PD2PD INIT

PD2PD:VOLT <NR2>

This command is used to set the voltage value in PD2.0 mode. (The set voltage value must be consistent with the voltage supported by the charger, otherwise an error will be reported)

Command Syntax

PD2PD:VOLT <NR2>

Parameters

<NR2>

Examples

PD2PD:VOLT 12

Query Syntax

PD2PD:VOLT?

Return Parameters

<NR2>

PD2PD:VOLT?

Query whether the voltage regulation is ready

Command Syntax

PD2PD:VOLT?

PD2PD:PDO:LIST:VOLT?

This command is used to query the PD2.0 output voltage list (Return Parameters: NRf)

Command Syntax

PD2PD:PDO:LIST:VOLT?

Parameters

5.000V,9.000V,12.000V

Examples

PD2PD:PDO:LIST:VOLT?

Return Parameters

<Nrf>V,<Nrf>V,..

PD2PD:PDO:LIST:CURREN?

This command is used to query the PD2.0 output current list

Command Syntax

PD2PD:PDO:LIST:CURREN?

Parameters

3.00A,3.00A,2.50A,2.00A

Examples

PD2PD:PDO:LIST:CURREN?

Return Parameters

<Nrf>A,<Nrf>A,..

PD2PD:CC SWIT

This command is used to switch the CC line of PD2.0 quick charge.

Command Syntax

PD2PD:CC SWIT

PD2PD:CC?

This command is used to query whether the CC line of PD2.0 is switched successfully

Command Syntax

PD2PD:CC?

Return Parameters

0: switching, 1: switching successfully, 2: switching failed

PD3PD INIT

This command is used to set the quick charge mode to PD3.0 (PPS) mode.

Command Syntax

PD3PD INIT

PD3PD:FIXED:VOLT <NR2>

This command is used to set the fixed point (POS) voltage value in PD3.0 mode. (The set voltage value must be consistent with the voltage supported by the charger, otherwise an error will be reported)

Command Syntax

```
PD3PD:FIXED:VOLT <NR2>
```

Parameters

<NR2>

Examples

```
PD3PD:FIXED:VOLT 15
```

Query Syntax

```
PD3PD:FIXED:VOLT?
```

Return Parameters

<BOOL>

PD3PD:FIXED:VOLT?

Query whether the voltage regulation is ready (BOOL)

Command Syntax

```
PD3PD:FIXED:VOLT?
```

Return Parameters

<BOOL>

PD3PD:PPSn:VOLT <Nr1>,<Nrf+>

This command is used to set the voltage value (PPS) of the nth point (adjustable voltage) in PD3.0 mode. (n must be an adjustable voltage group, otherwise an error will be reported)

Command Syntax

```
PD3PD:PPSn:VOLT <Nr1>,<Nrf+>
```

Parameters

MAX|MIN|MIN-MAX(The extreme value depends on the charger)

Examples

```
PD3PD:PPSn:VOLT <6>,< MAX|MIN|MIN-MAX>(The extreme value depends on the charger)
```

Query Syntax

PD3PD:PPSn:VOLT?

Return Parameters

<NR2>

PD3PD:PPSn:VOLT? <NR1>

This command is used to query the voltage value (PPS) of the nth point (adjustable voltage) in PD3.0 mode

Command Syntax

PD3PD:PPSn:VOLT?

Return Parameters

<NRf+>

PD3PD:PDO:LIST:VOLT?

This command is used to query the PD3.0 output voltage list.

Command Syntax

PD3PD:PDO:LIST:VOLT?

Return Parameters

<Nrf>V,<Nrf>V,<Nrf>V, <Nrf>V-<Nrf>V

PD3PD:PDO:LIST:CURR?

This command is used to query the PD3.0 output current list

Command Syntax

PD3PD:PDO:LIST:CURR?

Return Parameters

<Nrf>A,<Nrf>A,<Nrf>A

PD3PD:CC SWIT

This command is used to switch the CC line of PD3.0.

Command Syntax

PD3PD:CC SWIT

PD3PD:CC?

This command is used to query whether the CC line of PD3.0 is switched successfully

Command Syntax

PD3PD:CC?

Return Parameters

<0|1|2>

0: switching, 1: switching successfully, 2: switching failed

FCP INIT

This command is used to set the quick charge mode to FCP mode.

Command Syntax

FCP INIT

FCP:VOLT <Nrf+>

This command is used to set the voltage value in FCP mode.

(The set voltage value must be consistent with the voltage supported by the charger, otherwise an error will be reported)

Command Syntax

FCP:VOLT

Parameters

<Nrf+>

Examples

FCP:VOLT 9

Query Syntax

FCP:VOLT?

Return Parameters

<BOOL>

FCP:VOLT?

Query whether the voltage regulation is ready (BOOL)

Command Syntax

FCP:VOLT?

Return Parameters

<BOOL>

FCP:LIST:VOLT?

This command is used to query the output voltage list supported by the charger in FCP mode.

Command Syntax

FCP:LIST:VOLT?

FCP:LIST:CURR?

This command is used to query the output current list supported by the charger in FCP mode.

Command Syntax

FCP:LIST:CURR?

FCP:LIST:POW?

This command is used to query the maximum output power supported by the charger in FCP mode.

Command Syntax

FCP:LIST:POW?

Return Parameters

<Nrf>W

SCP INIT

This command is used to set the quick charge mode to SCP mode.

Command Syntax

SCP INIT

SCP:VOLT <NRf+>

This command is used to set the voltage value in SCP mode

Command Syntax

SCP:VOLT <NRf+>

Parameters

<NRf+>
MAX|MIN|MIN~MAX (The extreme value depends on the charger)

Query Syntax

SCP:VOLT?

Return Parameters

<NRf+>

SCP:VOLT?

This command is used to query the voltage value in SCP mode

Command Syntax

SCP:VOLT?

Return Parameters

<NRf+>

SCP:CURRE <NRf+>

This command is used to set the protection current value in SCP mode

Command Syntax

SCP:CURRE <NRf+>

Parameters

<NRf+>
MAX(5A)|MIN(0A)|MIN~MAX

Query Syntax

SCP:CURRE?

Return Parameters

<NRf+>

SCP:CURRE?

This command is used to query the protection current value in SCP mode

Command Syntax

SCP:CURRE?

Return Parameters

<NRf+>

SCP:LIST:VOLT?

This command is used to query the SCP voltage output list

Command Syntax

SCP:LIST:VOLT?

Return Parameters

<Nrf>V - <Nrf>V
For example, "5V-12V"

SCP:LIST:CURRE?

This command is used to query SCP OCP current

Command Syntax

SCP:LIST:CURRE?

Return Parameters

<Nrf>A

SCP:LIST:POW?

This command is used to query the maximum power of SCP

Command Syntax

SCP:LIST:POW?

Return Parameters

<Nrf>W

DPDN INIT

This command is used to initialize the DPDN mode (and clear the short-circuit state).

Command Syntax

DPDN INIT

DPDN DPSH

This command is used to short-circuit DP (output 3.3V to the DP pin).

Command Syntax

DPDN DPSH

DPDN DNSH

This command is used to short-circuit DN (input 3.3V to the DN pin).

Command Syntax

DPDN DNSH

DPDN:VOLT?

This command is used to query the DPDN voltage.

Command Syntax

DPDN:VOLT?

Return Parameters

<Nrf>V,<Nrf>V

For example, "1.200V, 3.200V"

QC:DISConnect <BOOL>

This command is used to disconnect the quick charge connection (disconnect the CC line, and D+ D- input high impedance).

Command Syntax

QC:DISConnect <BOOL>

Parameters

0 | 1 | OFF | ON

Query Syntax

QC:DISConnect?

Returned Parameters

0|1

Chapter 12 Auto-test Commands

PROGram:MAXimum:CURRent <NRf+>

This command is used to set the maximum current of the test

Command Syntax

PROGram:MAXimum:CURRent <NRf+>

Parameters

<NRf+>
0~MAX|MAX|MIN

Returned Parameters

<NRf+>

Examples

PROG:MAX:CURR 7

PROGram:MAXimum:CURRent?

This command is used to query the maximum current of the test

Command Syntax

PROGram:MAXimum:CURRent?

Parameters

None

Returned Parameters

<NRf+>

PROGram:MAXimum:VOLTage <NRf+>

This command is used to set the maximum voltage of the test

Command Syntax

PROGram:MAXimum:VOLTage <NRf+>

Parameters

<NRf+>
MIN~MAX|MAX|MIN

Returned Parameters

<NRf+>

Examples

PROG:MAX:VOLT 60

PROG:MAXimum:VOLTage?

This command is used to query the maximum voltage of the test

Command Syntax

PROG:MAXimum:VOLTage?

Parameters

None

Returned Parameters

<NRf+>

PROG:MAXimum:POWER <NRf+>

This command is used to set the maximum power of the test

Command Syntax

PROG:MAXimum:POWER <NRf+>

Parameters

<NRf+>

0~MAX |MAX|MIN

Returned Parameters

<NRf+>

Examples

PROG:MAX:POW 50

PROG:MAXimum:POWER?

This command is used to query the maximum power of the test

Command Syntax

PROG:MAXimum:POWER?

Parameters

None

Returned Parameters

<NRf+>

PROG:am:auT <boOL>

This command is used to set whether the test starts automatically

Command Syntax

PROG:am:auT <boOL>

Parameters

<boOL>

0 | 1 | OFF | ON

Returned Parameters

Not involved

Examples

PROG:am:auT 1

PROG:am:auT?

This command is used to query whether the test is automatically started

Command Syntax

PROG:am:auT?

Parameters

None

Returned Parameters

<boOL>

PROG:STARt:VoLTage <NRf+>

This command is used to set the starting voltage of the test

Command Syntax

PROG:STARt:VoLTage <NRf+>

Parameters

<NRf+>

0.1-MAX_VoLT|MAX|MIN

Examples

PROG:STAR:VoLT 3

PROGram:STARt:VOLTage?

This command is used to query the starting voltage of the test

Command Syntax

PROGram:STARt:VOLTage?

Parameters

None

Returned Parameters

<NRf+>

PROGram:BEEP <CRD>

This command is used to set whether the test beep

Command Syntax

PROGram:BEEP <CRD>

Parameters

<CRD>

PASS | FAIL

Returned Parameters

<CRD>

Examples

PROG:BEEP PASS

PROGram:BEEP?

This command is used to inquire whether the test is passed or failed to beep once.

Command Syntax

PROGram:BEEP?

Parameters

None

Returned Parameters

<CRD>

PROGram:OUTPut:CONDition <CRD>

This command is used to set the output conditions of the test

Command Syntax

PROG:OUTPut:CONDition <CRD>

Parameters

<CRD>

PASS | FAIL | END | DISable

Returned Parameters

<CRD>

Examples

PROG:OUTP:COND PASS

PROG:OUTPut:CONDition?

This command is used to query the output conditions of the test

Command Syntax

PROG:OUTPut:CONDition?

Parameters

None

Returned Parameters

< CRD>

PROG:OUTPut:MODE <CRD>

This command is used to set the output mode of the test

Command Syntax

PROG:OUTPut:MODE <CRD>

Parameters

<CRD>

LEVeI | PULSe

Returned Parameters

<CRD>

Examples

PROG:OUTP:MODE LEVEI

PROG:OUTPut:MODE?

This command is used to query the output mode of the test

Command Syntax

PROGram:OUTPut:MODE?

Parameters

None

Returned Parameters

<CRD>

PROGram:STOP:CONDition <CRD>

This command is used to set the stop condition of the test

Command Syntax

PROGram:STOP:CONDition <CRD>

Parameters

<CRD>

COMPlete | FAIL

Returned Parameters

<CRD>

Examples

PROGram:STOP:CONDition FALL

PROGram:STOP:CONDition?

This command is used to query the stop condition of the test

Command Syntax

PROGram:STOP:CONDition?

Parameters

None

Returned Parameters

<CRD>

PROGram:STEP:COUNT <NRf+>

This command is used to set the number of steps for the test

Command Syntax

PROGram:STEP:COUNT <NRf+>

Parameters

<NRf+>
2-50 | MAX | MIN

Returned Parameters

<NRf+>

Examples

PROG:STEP:COUNT 10

PROG:STEP:COUNT?

This command is used to query the number of steps for the test

Command Syntax

PROG:STEP:COUNT?

Parameters

None

Returned Parameters

< NRf+>

PROG[:STEP]:QCMODULE:PROTOCOL <NR1>,<CRD>

This command is used to set the test protocol selection

Command Syntax

PROG[:STEP]:QCMODULE:PROTOCOL <NR1>,<CRD>

Parameters

<NR1>, <CRD>
step,<NONE|QC2.0|QC3.0|PE1.0|PE2.0|PD2.0|PD3.0|FCP|SCP>

Returned Parameters

<CRD>

Examples

PROG:QCM:PROT 3, SCP

PROG[:STEP]:QCMODULE:PROTOCOL?<NR1>

This command is used to query the test protocol selection

Command Syntax

PROG[:STEP]:QCMODULE:PROTOCOL? <NR1>

Parameters

<NR1>

Returned Parameters

<CRD>

PROGram[:STEP]:QCModule:VSRC <NR1>,<NRf+>

This command is used to set the voltage of the test

Command Syntax

PROGram[:STEP]:QCModule:VSRC <NR1>,<NRf+>

Parameters

<NR1>, <NRf+>

step,<MIN to MAX (Depending on quick charge protocol) | MIN | MAX>

Returned Parameters

<NRf+>

Examples

PROG:QCM:VSRC 4,30

PROGram[:STEP]:QCModule:VSRC? <NR1>

This command is used to query the voltage of the test

Command Syntax

PROGram[:STEP]:QCModule:VSRC? <NR1>

Parameters

<NR1>

Returned Parameters

<NRf+>

PROGram[:STEP]:QCModule:ISRC <NR1>,<NRf+>

This command is used to set the test current

Command Syntax

PROGram[:STEP]:QCModule:ISRC <NR1>,<NRf+>

Parameters

<NR1>,<NRf+>

step,<0-5A | MIN | MAX>

Returned Parameters

<NRf+>

Examples

PROG:QCM:ISRC 3,3.2

PROGram[:STEP]:QCModule:ISRC? <NR1>

This command is used to query the test current

Command Syntax

PROGram[:STEP]:QCModule:ISRC? <NR1>

Parameters

<NR1>

Returned Parameters

<NRf+>

PROGram[:STEP]:QCModule:POS <NR1>,<NR1>

This command is used to set the working point of quick charging (applicable to PD and PPS)

Command Syntax

PROGram[:STEP]:QCModule:POS <NR1>,<NR1>

Parameters

<NR1>, <NR1>

1,2

Returned Parameters

POS<NR1>

Examples

PROG:QCM:POS 1,2

PROGram[:STEP]:QCModule:POS? <NR1>

This command is used to query the working point of quick charging (applicable to PD and PPS)

Command Syntax

PROGram[:STEP]:QCModule:POS? <NR1>

Parameters

<NR1>

Returned Parameters

POS<NR1>

PROGram[:STEP]:MODE <NR1>,<CRD>

This command is used to set the load mode during the test

Command Syntax

PROGram[:STEP]:MODE <NR1>,<CRD>

Parameters

<NR1>, <CRD>
step,<CC|CV|CR|CW>

Returned Parameters

<CC|CV|CR|CW>

Examples

PROG:MODE 2,CR

PROGram[:STEP]:MODE? <NR1>

This command is used to query the load mode during the test

Command Syntax

PROGram[:STEP]:MODE? <NR1>

Parameters

<NR1>

Returned Parameters

<CRD>

PROGram[:STEP]:SHORT <NR1>,<bool>

This command is used to set whether to perform short circuit test

Command Syntax

PROGram[:STEP]:SHORT <NR1>,<bool>

Parameters

<NR1>, <bool>
step,<0|1|OFF|ON>

Returned Parameters

<0|1|OFF|ON>

Examples

```
PROG:SHOR 1,ON
```

PROGram[:**STEP**]:**SHORT**? <NR1>

This command is used to query whether to perform short circuit test

Command Syntax

```
PROG
```

ram[:STEP]:SHORT? <NR1>

Parameters

<NR1>

Returned Parameters

<BOOL>

PROGram[:**STEP**]:**LEV**el <NR1>,<NRf+>

This command is used to set the level of each step of the test

Command Syntax

```
PROG
```

ram[:STEP]:LEVel <NR1>,<NRf+>

Parameters

<NR1>, <NRf+>

step,<MIN to MAX|MAX|MIN> (The extreme value depends on the load mode)

Returned Parameters

<NRf+>

Examples

```
PROG
```

ram:LEV 3,2.3

PROGram[:**STEP**]:**LEV**el? <NR1>

This command is used to query the level of each step of the test

Command Syntax

```
PROG
```

ram[:STEP]:LEVel? <NR1>

Parameters

<NR1>

Returned Parameters

<NRf+>

PROGram[:STEP]:READback <NR1>,<CRD>

This command is used to set the comparison item of the test

Command Syntax

```
PROGram[:STEP]:READback <NR1>,<CRD>
```

Parameters

<NR1>,<CRD>

step,<VOLTage|CURRent|POWER|VP+|VP-|VPP|IP+|IP-|IPP>

Returned Parameters

<CRD>

Examples

```
PROG:READ 5, CURRent
```

PROGram[:STEP]:READback? <NR1>

This command is used to query the comparison item of the test

Command Syntax

```
PROGram[:STEP]:READback? <NR1>
```

Parameters

<NR1>

Returned Parameters

<CRD>

<VOLTage|CURRent|POWER|VP+|VP-|VPP|IP+|IP-|IPP>

PROGram[:STEP]:READback:MAXimum <NR1>,<NRf+>

This command is used to set the upper limit of the comparison item

Command Syntax

```
PROGram[:STEP]:READback:MAXimum <NR1>,<NRf+>
```

Parameters

<NR1>,<NRf+>

step,<MIN to MAX|MAX|MIN> (The extreme value depends on the comparison item)

Returned Parameters

<NRf+>

Examples

PROG:READk:MAX 3,2.3

PROGrama[:STEP]:READback:MAXimum? <NR1>

This command is used to query the upper limit of the comparison item

Command Syntax

PROGrama[:STEP]:READback:MAXimum? <NR1>

Parameters

<NR1>

Returned Parameters

<NRf+>

PROGrama[:STEP]:READback:MINimum <NR1>,<NRf+>

This command is used to set the lower limit of the comparison item

Command Syntax

PROGrama[:STEP]:READback:MINimum <NR1>,<NRf+>

Parameters

<NR1>,<NRf+>

step,<MIN to MAX|MAX|MIN> (The extreme value depends on the comparison item)

Returned Parameters

<NRf+>

Examples

PROG:READk:MIN 3,2.3

PROGrama[:STEP]:READback:MINimum? <NR1>

This command is used to query the lower limit of the comparison item

Command Syntax

PROGrama[:STEP]:READback:MINimum? <NR1>

Parameters

<NR1>

Returned Parameters

<NRf+>

PROGram[:STEP]:DELay <NR1>,<NRf+>

This command is used to set the delay time of each step of the test

Command Syntax

PROGram[:STEP]:DELay <NR1>,<NRf+>

Parameters

<NR1>,<NRf+>
step,<0.1 to 30s|MAX|MIN>

Returned Parameters

<NRf+>

Examples

PROG:DEL 4,15

PROGram[:STEP]:DELay? <NR1>

This command is used to query the delay time of each step of the test

Command Syntax

PROGram[:STEP]:DELay? <NR1>

Parameters

<NR1>

Returned Parameters

<NRf+>

PROGram:SAVE <NR1>

This command is used to save the automatic test file.

Command Syntax

PROGram:SAVE <NR1>

Parameters

<NR1>
1~10

Returned Parameters

Not involved

Examples

PROG:SAVE 9

PROGram:RECall <NR1>

This command is used to recall the saved automatic test file.

Command Syntax

PROGram:RECall <NR1>

Parameters

<NR1>

1~10

Returned Parameters

Not involved

Examples

PROGram:RECall 9

PROGram[:STATe] <BOOL>

This command is used to set whether the test is run

Command Syntax

PROGram[:STATe] <BOOL>

Parameters

<BOOL>

0 | 1 | OFF | ON

Returned Parameters

<BOOL>

Examples

PROGram ON

PROGram[:STATe]?

This command is used to query whether the test is run

Command Syntax

PROGram[:STATe]?

Parameters

None

Returned Parameters

<BOOL>

PROGram:RUN:STEP?

This command is used to query which step the test is currently running at

Command Syntax

PROGram:RUN:STEP?

Parameters

None

Returned Parameters

<NR1>

Examples

PROG:RUN:STEP?

PROGram[:STEP]:RESult? <NR1>

This command is used to query whether the specified step test passed

Command Syntax

PROGram[:STEP]:RESult? <NR1>

Parameters

<NR1>

1-step

Examples

PROGram[:STEP]:RESult? 3

Returned Parameters

<BOOL>

Chapter 13 Battery Commands

BATTery:DISCharge:CURRent:LIMit <NRf+>

This command is used to set the current limit value in battery test mode

Command Syntax

BATTery:DISCharge:CURRent:LIMit <NRf+>

Parameters

<NRf+>

0 to rated value| MAX | MIN

Returned Parameters

<NRf+>

Examples

BATT:DISC:CURR:LIM 15

BATTery:DISCharge:CURRent:LIMit?

This command is used to query the current limit value in battery test mode

Command Syntax

BATTery:DISCharge:CURRent:LIMit?

Parameters

None

Returned Parameters

<NRf+>

BATTery:DISCharge:CURRent <NRf+>

This command is used to set the discharge current value in the battery test mode (that is, the load current setting value)

Command Syntax

BATTery:DISCharge:CURRent <NRf+>

Parameters

<NRf+>

0~LIMit | MAX | MIN

Returned Parameters

<NRf+>

Examples

BATT:DISC:CURR 1.2

BATTery:DISCharge:CURRent?

This command is used to query the discharge current value in the battery test mode (that is, the load current setting value)

Command Syntax

BATTery:DISCharge:CURRent?

Parameters

None

Returned Parameters

<NRf+>

BATTery:STOP:VOLTage <NRf+>

This command is used to set the stop voltage value of the battery test

Command Syntax

BATTery:STOP:VOLTage <NRf+>

Parameters

<NRf+>

0~MAX | MAX | MIN

Returned Parameters

<NRf+>

Examples

BATT:STOP:VOLT 20

BATTery:STOP:VOLTage?

This command is used to query the stop voltage value of the battery test

Command Syntax

BATTery:STOP:VOLTage?

Parameters

None

Returned Parameters

<NRf+>

BATTery:STOP:CAPacity <NRf+>

This command is used to set the cut-off capacity value of the battery test

Command Syntax

BATTery:STOP:CAPacity <NRf+>

Parameters

<NRf+>

MIN~MAX | MAX | MIN

Returned Parameters

<NRf+>

Examples

BATT:STOP:CAP 55.2

BATTery:STOP:CAPacity?

This command is used to query the cut-off capacity value of the battery test

Command Syntax

BATTery:STOP:CAPacity?

Parameters

None

Returned Parameters

<NRf+>

BATTery:STOP:TIME <NRf+>

This command is used to set the termination time of the battery test

Command Syntax

BATTery:STOP:TIME <NRf+>

Parameters

<NRf+>

MIN~MAX | MAX | MIN

Returned Parameters

<NRf+>

Examples

BATT:STOP:TIME 1000

BATTery:STOP:TIME?

This command is used to query the termination time of the battery test

Command Syntax

BATTery:STOP:TIME?

Parameters

None

Returned Parameters

<NRf+>

BATTery:SAVE <NRf+>

This command is used to save the set battery test data to E2PROM

Command Syntax

BATTery:SAVE <NRf+>

Parameters

<NRf+>

1~10|MIN|MAX

Returned Parameters

Not involved

Examples

BATT:SAVE 5

BATTery:RECall <NRf+>

This command is used to recall the test file of the specified group from the storage

Command Syntax

BATTery:RECall <NRf+>

Parameters

<NRf+>

1~10|MIN|MAX

Returned Parameters

Not involved

BATTery[:STATe] <BOOL>

This command is used to turn on or off the battery test state

Command Syntax

BATTery[:STATe] <BOOL>

Parameters

<BOOL>
0 | 1 | OFF | ON

Returned Parameters

0 | 1 | OFF | ON

Examples

BATT ON

BATTery[:STATe]?

This command is used to query the on or off status of the battery test

Command Syntax

BATTery[:STATe]?

Parameters

None

Returned Parameters

0 | 1 | OFF | ON

BATTery:TIME?

This command is used to query the battery test time

Command Syntax

BATTery:TIME?

Parameters

None

Returned Parameters

<NR1>

BATTery:CAPacity?

This command is used to query the discharged capacity

Command Syntax

BATTery:CAPacity?

Parameters

None

Returned Parameters

<NR2>

Chapter 14 OCP Test Commands

OCP:VON <NRf+>

This command is used to set the starting load voltage value of OCP test

Command Syntax

OCP:VON <NRf+>

Parameters

<NRf+>

0~MAX| MAX | MIN

Returned Parameters

<NRf+>

Examples

OCP:VON 33

OCP:VON?

This command is used to query the starting load voltage value of OCP test

Command Syntax

OCP:VON?

Parameters

None

Returned Parameters

<NRf+>

OCP:VON:DElay <NRf+>

This command is used to set the load delay of OCP test

Command Syntax

OCP:VON:DElay <NRf+>

Parameters

<NRf+>

0~99.9s | MAX | MIN

Returned Parameters

<NRf+>

Examples

OCP:VON:DEL 45.0

OCP:VON:DELaY?

This command is used to query the load delay of OCP test

Command Syntax

OCP:VON:DELaY?

Parameters

None

Returned Parameters

<NRf+>

OCP:CURRent:Range <NRf+>

This command is used to set the current range of OCP test

Command Syntax

OCP:CURRent:Range <NRf+>

Parameters

<NRf+>

0~MAX | MAX | MIN

Returned Parameters

<NRf+>

Examples

OCP:CURRent:Range 15

OCP:CURRent:Range?

This command is used to query the current range of OCP test

Command Syntax

OCP:CURRent:Range?

Parameters

None

Returned Parameters

<NRf+>

OCP:ISart <NRf+>

This command is used to set the initial current of OCP test

Command Syntax

OCP:ISart <NRf+>

Parameters

<NRf+>

0~Range | MAX | MIN

Returned Parameters

<NRf+>

Examples

OCP:IST 0

OCP:ISart?

This command is used to query the initial current of OCP test

Command Syntax

OCP:ISart?

Parameters

None

Returned Parameters

<NRf+>

OCP:IEND <NRf+>

This command is used to set the cut-off current of OCP test

Command Syntax

OCP:IEND <NRf+>

Parameters

<NRf+>

0~Range | MAX | MIN

Returned Parameters

<NRf+>

Examples

OCP:IEND 12

OCP:IEND?

This command is used to query the cut-off current of OCP test

Command Syntax

OCP:IEND?

Parameters

None

Returned Parameters

<NRf+>

OCP:STEP <NRf+>

This command is used to set the step current of OCP test

Command Syntax

OCP:STEP <NRf+>

Parameters

<NRf+>

0~Range | MAX | MIN

Returned Parameters

<NRf+>

Examples

OCP:STEP 0.2

OCP:STEP?

This command is used to query the step current of OCP test

Command Syntax

OCP:STEP?

Parameters

None

Returned Parameters

<NRf+>

OCP:DWELI <NRf+>

This command is used to set the duration of each step of the OCP test

Command Syntax

OCP:DWELI <NRf+>

Parameters

<NRf+>

0.1~99.9s | MAX | MIN

Returned Parameters

<NRf+>

Examples

OCP:DWELI 20

OCP:DWELI?

This command is used to query the duration of each step of the OCP test

Command Syntax

OCP:DWELI?

Parameters

None

Returned Parameters

<NRf+>

OCP:VTRig <NRf+>

This command is used to set the comparison voltage of OCP test

Command Syntax

OCP:VTRig <NRf+>

Parameters

<NRf+>

0~MAX| MAX | MIN

Returned Parameters

<NRf+>

Examples

OCP:VTRig 30

OCP:VTRig?

This command is used to query the comparison voltage of OCP test

Command Syntax

OCP:VTRig?

Parameters

None

Returned Parameters

<NRf+>

OCP:MIN:TRIP <NRf+>

This command is used to set the lower limit of the judgment current of OCP test

Command Syntax

OCP:MIN:TRIP <NRf+>

Parameters

<NRf+>

ISart~~IEND | MAX | MIN

Returned Parameters

<NRf+>

Examples

OCP:MIN:TRIP 7.5

OCP:MIN:TRIP?

This command is used to query the lower limit of the judgment current of OCP test

Command Syntax

OCP:MIN:TRIP?

Parameters

None

Returned Parameters

<NRf+>

OCP:MAX:TRIP <NRf+>

This command is used to set the upper limit of the judgment current of OCP test

Command Syntax

OCP:MAX:TRIP <NRf+>

Parameters

<NRf+>

ISart~~IEND | MAX | MIN

Returned Parameters

<NRf+>

Examples

OCP:MAX:TRIP 18.0

OCP:MAX:TRIP?

This command is used to query the upper limit of the judgment current of OCP test

Command Syntax

OCP:MAX:TRIP?

Parameters

None

Returned Parameters

<NRf+>

OCP:SAV <NR1>

This command is used to save the edited OCP test data to the memory

Command Syntax

OCP:SAV <NR1>

Parameters

<NR1>

1~10

Returned Parameters

Not involved

Examples

OCP:SAV 8

OCP:RCL <NR1>

This command is used to recall the OCP test from the memory

Command Syntax

OCP:RCL <NR1>

Parameters

<NR1>

Returned Parameters

Not involved

Examples

OCP:RCL 8

OCP[:STATe] <BOOL>

This command is used to turn on or off the OCP test state

Command Syntax

OCP[:STATe] <BOOL>

Parameters

<BOOL>

0 | 1 | OFF | ON

Returned Parameters

<BOOL>

Examples

OCP ON

OCP[:STATe]?

This command is used to query the on or off status of OCP test

Command Syntax

OCP[:STATe]?

Parameters

None

Returned Parameters

<BOOL>

OCP:RESult[:OCP]?

This command is used to query OCP test results

Command Syntax

OCP:RESult[:OCP]?

Parameters

None

Returned Parameters

Testing: -1;

Did not enter the OCP protection state: -2;

Positive value: OCP value

Examples

OCP:RES?

OCP:RESult:PMAX?

This command is used to query the maximum power point during OCP test

Command Syntax

OCP:RESult:PMAX?

Parameters

None

Returned Parameters

The return order is W, V, A

Chapter 15 OPP Test Commands

OPP:VON <NRf+>

This command is used to set the starting load voltage value of OPP test

Command Syntax

OPP:VON <NRf+>

Parameters

<NRf+>

0~MAX | MAX | MIN

Returned Parameters

<NRf+>

Examples

OPP:VON 60

OPP:VON?

This command is used to query the starting load voltage value of OPP test

Command Syntax

OPP:VON?

Parameters

None

Returned Parameters

<NRf+>

OPP:VON:DELAy <NRf+>

This command is used to set the load delay of OPP test

Command Syntax

OPP:VON:DELAy <NRf+>

Parameters

<NRf+>

0~99.9s | MAX | MIN

Returned Parameters

<NRf+>

Examples

OPP:VON:DEL 60.0

OPP:VON:DELaY?

This command is used to query the load delay of OPP test

Command Syntax

OPP:VON:DELaY?

Parameters

None

Returned Parameters

<NRf+>

OPP:CURRent:Range <NRf+>

This command is used to set the current range of OPP test

Command Syntax

OPP:CURRent:Range <NRf+>

Parameters

<NRf+>

0~MAX | MAX | MIN

Returned Parameters

<NRf+>

Examples

OPP:CURR:Range 18

OPP:CURRent:Range?

This command is used to query the current range of OPP test

Command Syntax

OPP:CURRent:Range?

Parameters

None

Returned Parameters

<NRf+>

OPP:PStart <NRf+>

This command is used to set the initial power of OPP test

Command Syntax

OPP:PStart <NRf+>

Parameters

<NRf+>

0 to rated value | MAX | MIN

Returned Parameters

<NRf+>

Examples

OPP:PST 0

OPP:PStart?

This command is used to query the initial power of OPP test

Command Syntax

OPP:PStart?

Parameters

None

Returned Parameters

<NRf+>

OPP:PEND <NRf+>

This command is used to set the cut-off power of OPP test

Command Syntax

OPP:PEND <NRf+>

Parameters

<NRf+>

0 to rated value | MAX | MIN

Returned Parameters

<NRf+>

Examples

OPP:PEND 125

OPP:PEND?

This command is used to query the cut-off power of OPP test

Command Syntax

OPP:PEND?

Parameters

None

Returned Parameters

<NRf+>

OPP:STEP <NRf+>

This command is used to set the step power of OPP test

Command Syntax

OPP:STEP <NRf+>

Parameters

<NRf+>

0~MAX| MAX | MIN

Returned Parameters

<NRf+>

Examples

OPP:STEP 2

OPP:STEP?

This command is used to query the step power of OPP test

Command Syntax

OPP:STEP?

Parameters

None

Returned Parameters

<NRf+>

OPP:DWELI <NRf+>

This command is used to set the duration of each step of the OPP test

Command Syntax

OPP:DWELI <NRf+>

Parameters

<NRf+>

0.1~99.9s | MAX | MIN

Returned Parameters

<NRf+>

Examples

OPP:DWELI 2

OPP:DWELI?

This command is used to query the duration of each step of the OPP test

Command Syntax

OPP:DWELI?

Parameters

None

Returned Parameters

<NRf+>

OPP:VTRig <NRf+>

This command is used to set the comparison voltage of OPP test

Command Syntax

OPP:VTRig <NRf+>

Parameters

<NRf+>

MIN~MAX | MAX | MIN

Returned Parameters

<NRf+>

Examples

OPP:VTR 66

OPP:VTRig?

This command is used to query the comparison voltage of OPP test

Command Syntax

OPP:VTRig?

Parameters

None

Returned Parameters

<NRf+>

OPP:MIN:TRIP <NRf+>

This command is used to set the lower limit of judgment power for OPP test

Command Syntax

OPP:MIN:TRIP <NRf+>

Parameters

<NRf+>

PStart~PEND | MAX | MIN

Returned Parameters

<NRf+>

Examples

OPP:MIN:TRIP 1

OPP:MIN:TRIP?

This command is used to query the lower limit of judgment power for OPP test

Command Syntax

OPP:MIN:TRIP?

Parameters

None

Returned Parameters

<NRf+>

OPP:MAX:TRIP <NRf+>

This command is used to set the upper limit of the judgment power of OPP test

Command Syntax

OPP:MAX:TRIP <NRf+>

Parameters

<NRf+>

PStart~PEND | MAX | MIN

Returned Parameters

<NRf+>

Examples

OPP:MAX:TRIP 80

OPP:MAX:TRIP?

This command is used to query the upper limit of the judgment power of OPP test

Command Syntax

OPP:MAX:TRIP?

Parameters

None

Returned Parameters

<NRf+>

OPP:SAV <NR1>

This command is used to save the edited OPP test data to the memory

Command Syntax

OPP:SAV <NR1>

Parameters

<NR1>

1~10

Returned Parameters

Not involved

Examples

OPP:SAV 9

OPP:RCL <NR1>

This command is used to recall the OPP test from the memory

Command Syntax

OPP:RCL <NR1>

Parameters

<NR1>

1~10

Returned Parameters

Not involved

Examples

OPP:RCL 9

OPP[:STATe] <BOOL>

This command is used to turn on or off the OPP test state

Command Syntax

OPP[:STATe] <BOOL>

Parameters

<BOOL>

0 | 1 | OFF | ON

Returned Parameters

<BOOL>

Examples

OPP ON

OPP[:STATe]?

This command is used to query the on or off status of OPP test

Command Syntax

OPP[:STATe]?

Parameters

None

Returned Parameters

0 | 1 | OFF | ON

OPP:RESult[:OPP]?

This command is used to query OPP test results

Command Syntax

OPP:RESult[:OPP]?

Parameters

None

Returned Parameters

Testing: -1;

Did not enter the OPP protection state: -2;

Positive value: OPP value

OPP:RESult:PMAX?

This command is used to query the information of the maximum power point during the OPP test

Command Syntax

OPP:RESult:PMAX?

Parameters

None

Returned Parameters

The return order is W, V, A

Chapter 16 Trace Commands

TRACe:CLEAr

This command is used to clear the reading buffer

Command Syntax

TRACe:CLEAr

TRACe:FREE?

This command is used to query the status of the memory

Command Syntax

TRACe:FREE?

Parameters

None

Returned Parameters

<NR1, NR1> (To be cached, cached)

TRACe:POINts <NRf+>

This command is used to specify the size of the cache

Command Syntax

TRACe:POINts <NRf+>

Parameters

<NRf+>

2~2000 | MINimum | MAXimum | DEFault(2000)

Default value

2000

Returned Parameters

<NRf+>

Examples

TRAC:POIN 1000

TRACe:POINts?

This command is used to query the size of the cache

Command Syntax

TRACe:POINts?

Parameters

None

Returned Parameters

<NRf+>

TRACe:FEED[:SElected] <CRD>

This command is used to select the data source to put in the cache

Command Syntax

TRACe:FEED[:SElected] <CRD>

Parameters

<CRD>

VOLTage|CURRent|TWO(Reset)

Returned Parameters

<CRD>

Examples

TRAC:FEED VOLTage

TRACe:FEED[:SElected]?

This command is used to query the data source

Command Syntax

TRACe:FEED[:SElected]?

Parameters

None

Returned Parameters

<CRD>

TRACe:FEED:CONTRol <CRD>

This command is used to select cache control parameters

Command Syntax

TRACe:FEED:CONTRol <CRD>

Parameters

<CRD>
NEVer (Reset) | NEXT

Returned Parameters

<CRD>

Examples

TRAC:FEED:CONT NEXT

TRACe:FEED:CONTRol?

This command is used to query the cache control parameters

Command Syntax

TRACe:FEED:CONTRol?

Parameters

None

Returned Parameters

<CRD>

TRACe:FILTer[:STATe] <BOOL>

This command is used to select whether to filter the data

Command Syntax

TRACe:FILTer[:STATe] <BOOL>

Parameters

<BOOL>
(Reset)0 |1|OFF|ON

Returned Parameters

<BOOL>

Examples

TRAC:FILT ON

TRACe:FILTer[:STATe]?

This command is used to query whether to filter the data

Command Syntax

TRACe:FILTer[:STATe]?

Parameters

None

Returned Parameters

<BOOL>

TRACe:DELAy <NRf+>

This command is used to set the buffer trigger delay time

Command Syntax

TRACe:DELAy <NRf+>

Parameters

<NRf+>

0 to 3600s | MINimum | MAXimum | DEFault (0S)

Returned Parameters

<NRf+>

Examples

TRACe:DELAy 2

TRACe:DELAy?

This command is used to query the buffer trigger delay time

Command Syntax

TRACe:DELAy?

Parameters

None

Returned Parameters

<NRf+>

TRACe:TIMer <NRf+>

This command is used to set the cache interval

Command Syntax

TRACe:TIMer <NRf+>

Parameters

<NRf+>

20us~3600s | MIN | MAX | DEF(1S)

Returned Parameters

<NRf+>

Examples

TRACe:TIMer 30

TRACe:TIMer?

This command is used to query the cache interval

Command Syntax

TRACe:TIMer?

Parameters

None

Returned Parameters

<NRf+>

TRACe:DATA?

This command is used to send the reading value stored in the buffer to the computer

Command Syntax

TRACe:DATA?

Chapter 17 Peak Commands

The Peak command can be used to read the maximum/minimum value after starting the test. After starting the Peak test, the peak record is automatically cleared.

PEAK[:STATe] <bool>

This command is used to set the peak test status.

Command Syntax

PEAK[:STATe] <bool>

Parameters

0 | 1 | OFF | ON

Examples

Peak:ON

Query Syntax

0 PEAK[:STATe]?

Returned Parameters

0 | 1

PEAK CLEAr

This command is used to clear the peak record.

Command Syntax

PEAK CLEAr

Examples

Peak CLE

PEAK:VOLTage:MAXimum?

This command is used to query the maximum voltage.

Command Syntax

PEAK:VOLTage:MAXimum?

Examples

PEAK:VOLT:MAX?

Returned Parameters

<NR2>

PEAK:VOLTage:MINimum?

This command is used to query the minimum voltage.

Command Syntax

PEAK:VOLTage:MINimum?

Examples

PEAK:VOLT:MIN?

Returned Parameters

<NR2>

PEAK:CURRent:MAXimum?

This command is used to query the maximum current.

Command Syntax

PEAK:CURRent:MAXimum?

Examples

PEAK: CURR:MAX?

Returned Parameters

<NR2>

PEAK:CURRent:MINimum?

This command is used to query the minimum current.

Command Syntax

PEAK: CURRent:MINimum?

Examples

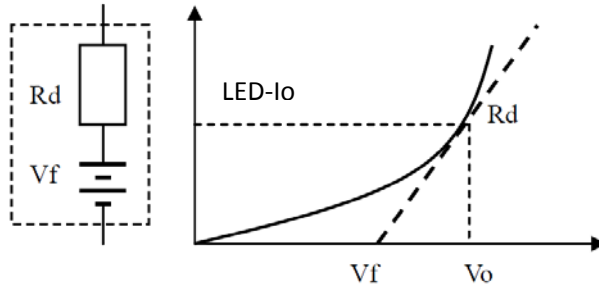
PEAK:CURR:MIN?

Returned Parameters

<NR2>

Chapter 18 CR-LED Commands

IT8500G+ series electronic load provides LED simulation function. The LED equivalent circuit is shown in the figure below. It is the series connection of resistor R_d and voltage source V_f . Its VI curve is equivalent to the tangent of the real LED nonlinear VI curve at the operating point (V_o , LED- I_o).



Users need to set 3 parameters: V_o , LED- I_o , R_d -Coeff:

- LED- I_o represents the rated output current of the tested LED power supply;
- V_o represents the forward operating voltage of the LED at the forward operating current LED- I_o , which can be obtained from the VI curve in the LED specification. The real load is usually connected with n LEDs in series, so V_o should be set to n times the single-cell parameter, Can also be set to any value within the output voltage range of the LED power supply;
- R_d -Coeff is the ratio of the series equivalent resistance (R_d) to the total equivalent resistance of the LED load ($V_o/\text{LED-}I_o$), that is, $R_d\text{-Coeff} = R_d/(V_o/\text{LED-}I_o)$. In series applications, R_d - The Coeff parameter is only related to the VI curve of the selected LED itself, and has nothing to do with the number of series connections.

Therefore, the user can determine the LED- I_o according to the rated output current of the LED power supply, and then calculate the R_d -Coeff parameter in combination with the VI curve on the LED specification, and then adjust arbitrarily within the output voltage range of the LED power supply to realize the real LED simulated pull test for the LED power supply.

LED:VOLTage

This command is used to set LED V_o .

Command Syntax

LED:VOLTage <Nrf+>

Parameters

0~MAX|MIN|MAX|DEF(150V)

Examples

LED:VOLT 18

Query Syntax

LED:VOLT?

Return Parameters

<NR2>

LED:CURRent

This command is used to set the LED Io.

Command Syntax

LED:CURRent <Nrf+>

Parameters

0~C-RANGE|MIN|MAX|DEF (0A)

Examples

LED:CURR 0.35

Query Syntax

LED:CURR?

Return Parameters

<NR2>

LED:RCOeff

This command is used to set LED Rd Coeff.

Command Syntax

LED:RCOeff <Nrf+>

Parameters

0.001~0.999|MIN|MAX|DEF(0.7)

Examples

LED:RCO 0.2

Query Syntax

LED:RCO?

Return Parameters

<NR2>

Chapter 19 IEEE-488 Commands

This chapter describes IEEE-488 common commands provided by the IT8500G+ series electronic load.

***CLS - Clear Status**

This command clears the registers:

- _ Standard Event Register
- _ Operation Event Register
- _ Questionable Event Register
- _ Error Queue

Command syntax

*CLS

Parameters

None

***ESE <NRf> - Event Enable**

This command programs the Standard Event Status Enable register bits. The programming determines which events of the Standard Event Status Event register (see *ESR?) are allowed to set the ESB (Event Summary Bit)

of the Status Byte register. A "1" in the bit position enables the corresponding event. All of the enabled events of the Standard Event Status Event Register are logically ORed to cause the Event Summary Bit (ESB) of the Status Byte Register to be set. See chapter "Programming the Status Registers" for descriptions of the Standard Event Status registers.

The query reads the Standard Event Status Enable register.

Command Syntax

*ESE <NRf>

Parameters

0 to 255

Powe-on Value

see *PSC

Examples

*ESE 129

Query Syntax

*ESE?

Returned Parameters

<NR1>

Related Commands

*ESR? *PSC *STB?

*ESE?

This query reads the Standard Event Enable register.

Command Syntax

*ESE <NRf>

Returned Parameters

0 to 255

Default value

see *PSC

*ESR?

This query reads the Standard Event Status Event register. Reading the register clears it. The bit configuration of this register is the same as the Standard Event Status Enable register (see *ESE). See chapter “Programming the Status Registers” for a detailed explanation of this register.

Query Syntax

*ESR?

Parameters

None

Returned Parameters

<NR1> (register value)

Related Commands

*CLS *ESE *ESE? *OPC

*IDN?

This query requests the electronic load to identify itself. It returns the data in four fields separated by comma.

Query Syntax

*IDN?

Parameters

None

Examples

ITECH Ltd, IT85XX, XXXXXXXXXXXXXXXXXXXXX, 1.21-1.28

Returned Parameters

<AARD> Field Information

Technologies Manufacturer

ITECH Ltd

Model

IT85XX

Serial Number

XXXX.....

Firmware Revision

1.21-1.28

*OPC

This command causes the interface to set the OPC bit (bit 0) of the Standard Event Status register when the electronic load has completed all pending operations. (See *ESE for the bit configuration of the Standard Event Status registers.) Pending operations are complete when:

- All commands sent before *OPC have been executed. This includes overlapped commands. Most commands are sequential and are completed before the next command is executed. Overlapped commands are executed in parallel with other commands. Commands that affect trigger actions are overlapped with subsequent commands sent to the electronic load. The *OPC command provides notification that all overlapped commands have been completed.
- All triggered actions are completed and the trigger system returns to the Idle state.

*OPC does not prevent processing of subsequent commands but Bit 0 will not be set until all pending operations are completed. The query causes the interface to place an ASCII "1" in the Output Queue when all pending operations are completed.

Command Syntax

*OPC

Parameters

None

Query Syntax

*OPC?

Returned Parameters

<NR1>

Related Commands

*TRIG *WAI

*OPC?

This command is used to query whether the command execution is completed or not.

Command Syntax

*OPC?

Parameters

None

*PSC

This command is used to control whether the electronic load will generate a service request when power on again.

1 OR ON: When the load power on, status byte enable register, operater event enable register, query event enable register and standard event enable register will be cleared.

0 OR OFF: The value of status byte enable register, operater event enable register, query event enable register and standard event enable register will be stored in the None-volatile storage, which will be recalled when power on.

Command Syntax

*PSC <bool>

Parameters

0|1|ON|OFF

Query Syntax

*PSC?

Returned Parameters

0|1

*PSC?

This command is used to query the status byte enable register.

Command Syntax

*PSC?

Returned Parameters

0|1|ON|OFF

*RCL

This command restores the electronic load to a state that was previously stored in memory with a *SAV command to the specified location. All states are recalled with the following exceptions:

CAL:STATE is set to OFF

The trigger system is set to the Idle state by an implied ABORt command (this cancels any uncompleted trigger actions)

NOTE: The device state stored in location 0 is automatically recalled at power turn-on.

Command Syntax

*RCL <NRf>

Parameters

0 to 9

Examples

*RCL 3

Related Commands

*PSC *RST *SAV

*RST

This command reset the electronic load to the factory-defined states.

Command Syntax

*RST

Parameters

None

*SAV

This command stores the present state of the electronic load to a specified location in memory. Up to 100 states can be stored. If a particular state is desired at power-on, it should be stored in location 0. It then will be recalled at power-on if the power-on state is set to RCL0. Use *RCL to retrieve instrument states.

Command Syntax

*SAV <NRf>

Parameters

0 - 99

Examples

*SAV 3

Related Commands

*PSC *RST *RCL

*SRE

This command sets the condition of the Service Request Enable Register. This register determines which bits from the Status Byte Register (see *STB for its bit configuration) are allowed to set the Master Status Summary (MSS) bit and the Request for Service (RQS) summary bit. A 1 in any Service Request Enable Register bit position enables the corresponding Status Byte Register bit and all such enabled bits then are logically ORed to cause Bit 6 of the Status Byte Register to be set.

When the controller conducts a serial poll in response to SRQ, the RQS bit is cleared, but the MSS bit is not. When *SRE is cleared (by programming it with 0), the electronic load cannot generate an SRQ to the controller. The query returns the current state of *SRE.

Command Syntax

*SRE <NRf>

Parameters

0 to 255

Default value

see *PSC

Examples

*SRE 128

Query Syntax

*SRE?

Returned Parameters

<NR1> (register binary value)

Related Commands

*ESE *ESR *PSC

*SRE?

This command is used to query status byte group enable register.

Command Syntax

*SRE?

Returned Parameters

0 - 255

Default value

see *PSC

***STB?**

This query reads the Status Byte register, which contains the status summary bits and the Output Queue MAV bit. Reading the Status Byte register does not clear it. The input summary bits are cleared when the appropriate event registers are read (see chapter “Programming the Status Registers” for more information). A serial poll also returns the value of the Status Byte register, except that bit 6 returns Request for Service (RQS) instead of Master Status Summary (MSS). A serial poll clears RQS, but not MSS. When MSS is set, it indicates that the electronic load has one or more reasons for requesting service.

Query Syntax

*STB?

Parameters

None

Returned Parameters

<NR1> (register value)

Related Commands

*SRE *ESR *ESE

***TST?**

This command causes the electronic load to do a self-test and report any error.

The reference information for the return value is as follows:

0: indicates no error

1: indicates that the module failed to initialize.

3: indicates that the module calibration data is lost.

4 or 5: indicates EEPROM error

Command Syntax

TST?

Parameters

None

Returned Parameters

<NR1> 0 indicates the electronic load has passed selftest.

Non-zero indicates an error code.

***TRG**

This command generates a trigger to any system that has BUS selected as its

source.

Chapter 20 Error Messages

Error Number List

This chapter gives the error numbers and descriptions that are returned by the electronic load. Error numbers are returned in two ways:

- Error numbers are displayed on the front panel
- Error numbers and messages are read back with the `SYSTEM:ERRor?` query. `SYSTEM:ERRor?` returns the error number into a variable and returns two parameters, an NR1 and a string.

The following table lists the errors that are associated with SCPI syntax errors and interface problems. It also lists the device dependent errors. Information inside the brackets is not part of the standard error message, but is included for clarification. When errors occur, the Standard Event Status register records them in bit 2, 3, 4, or 5:

Command Errors 100 through 199 (sets Standard Event Status Register bit #5 CME)

Error	Error String
101	device independed error Too many numeric suffices in Command Spec
110	No Input Command to parse
114	Numeric suffix is invalid value
116	Invalid value in numeric or channel list, e.g. out of range
117	Invalid number of dimensions in a channel list
120	Parameter of type Numeric Value overflowed its storage
130	Wrong units for parameter
140	Wrong type of parameter(s)
150	Wrong number of parameters
160	Unmatched quotation mark (single/double) in parameters
165	Unmatched bracket
170	Command keywords were not recognized
180	No entry in list to retrieve (number list or channel list)
190	Too many dimensions in entry to be returned in parameters
191	Too many char

Execution Errors –200 through –299 (sets Standard Event Status Register bit #4 EXE)

Error	Error String
–200	Execution error [generic]
–221	Settings conflict [check current device state]
–222	Data out of range [e.g., too large for this device]

-223	Too much data [out of memory; block, string, or expression too long]
-224	Illegal parameter value [device-specific]
-225	Out of memory
-270	Macro error
-272	Macro execution error
-273	Illegal macro label
-276	Macro recursion error
-277	Macro redefinition not allowed

System Errors –300 through –399 (sets Standard Event Status Register bit #3 DDE)

Error	Error String
-310	System error [generic]
-350	Too many errors [errors beyond 9 lost due to queue overflow]

Query Errors "C400 through "C499 (sets Standard Event Status Register bit #2)

Error	Error String
-400	Query error [generic]
-410	Query INTERRUPTED [query followed by DAB or GET before response complete]
-420	Query UNTERMINATED [addressed to talk, incomplete programming message received]
-430	Query DEADLOCKED [too many queries in command string]
-440	Query UNTERMINATED [after indefinite response]

Selftest Errors 0 through 99 (sets Standard Event Status Register bit #3)

Error	Error String
0	No error
1	Module Initialization Lost
2	Mainframe Initialization Lost
3	Module Calibration Lost
4	Non-volatile RAM STATE section checksum failed
5	Non-volatile RAM RST section checksum failed
10	RAM selftest
11	CVDAC selftest 1
12	CVDAC selftest 2
13	CCDAC selftest 1
14	CCDAC selftest 2
15	CRDAC selftest 1
16	CRDAC selftest 2
20	Input Down

40	Flash write failed
41	Flash erase failed
80	Digital I/O selftest error

Device-Dependent Errors 100 through 32767 (sets Standard Event Status Register bit #3)

Error	Error String
213	RS-232 buffer overrun error
216	RS-232 receiver framing error
217	RS-232 receiver parity error
218	RS-232 receiver overrun error
220	Front panel uart overrun
221	Front panel uart framing
222	Front panel uart parity
223	Front panel buffer overrun
224	Front panel timeout
225	Front Crc Check error
226	Front Cmd Error
401	CAL switch prevents calibration
402	CAL password is incorrect
403	CAL not enabled
404	Computed readback cal constants are incorrect
405	Computed programming cal constants are incorrect
406	Incorrect sequence of calibration commands
407	CV or CC status is incorrect for this command
603	FETCH of data that was not acquired
604	Measurement overrange

Contact Us

Thanks for purchasing ITECH products. In case of any doubts, please contact us as follows:

1. Visit ITECH website: www.itechate.com.
2. Select the most convenient contact method for further information.