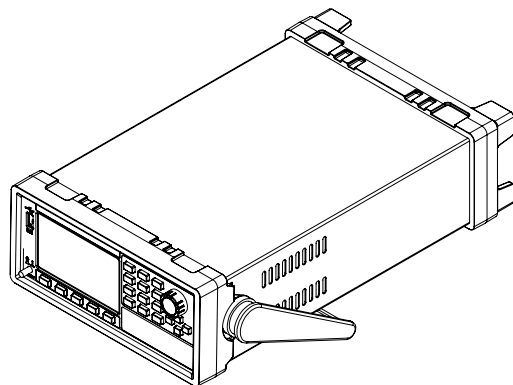


Power meter

IT9120 Series Programming Guide



Notices

© ItechElectronic, Co., Ltd. 2020
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

IT9120-402146

Revision

3rd Edition: May 11, 2020
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

U.S. Government Restricted Rights. Software and technical data rights granted to the federal government include only those rights customarily provided to end user customers. ITECH provides this customary commercial license in software and technical data pursuant to FAR 12.211 (Technical Data) and 12.212 (Computer Software) and DFARS 252.227-7015 (Technical Data – Commercial Items) and DFARS 227.7202-3 (Rights in Commercial Computer Software or Computer Software Documentation).

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 IT9120 series power meter meets all the published specifications.

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 if the product is:

- Damaged resulting from customer-wired circuits or customer-supplied parts or accessories.
- Modified or repaired by customer without authorization.
- Damaged resulting from customer-wired circuits or use in an environment not designated by us;
- 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)
	Alternating current		OFF (power)
	Both direct and alternating current		Power-on state
	Protective earth (ground) terminal		Power-off state
	Earth (ground) terminal		Reference terminal
	Caution		Positive terminal
	Warning (refer to this manual for specific Warning or Caution information)		Negative terminal
	A 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

- Do not use the instrument if it is damaged. Before operation, check the casing to see whether it cracks or is not applied with sufficient plastics. Do not operate the instrument in the presence of inflammable gasses, vapors or dusts.
- Make sure to use the power cord supplied by ITECH.
- Check all marks on the instrument before connecting the instrument to power supply.
- Turn off the instrument and the operation system before connecting to the I/O terminal.
- Do not use the instrument if the detachable cover is removed or loosen.
- Do not connect the instrument to any cable or terminal block before self-testing.
- To prevent the possibility of accidental injuries, be sure to use the power adapter supplied by the manufacturer only.
- 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.




Environmental Conditions	Requirements
Operating temperature	5°C-40°C
Operating humidity	humidity 20%-80% (non-condensation)
Storage temperature	-20°C-50 °C
Altitude	Operating up to 2,000 meters
Installation category	II
Pollution degree	Pollution degree 2



NOTE

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

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 affix 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 useful 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 1 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.....	ii
Regulatory Markings	iii
Waste Electrical and Electronic Equipment (WEEE) Directive	iii
Compliance Information.....	iv
Chapter1 Remote Control	1
1.1 Overview	1
1.2 Command type	1
1.3 Command format	3
1.4 Data type	4
1.5 Remote Operation	5
Chapter2 Status Register Commands	8
Standard Event Register	8
Status Byte Register.....	8
STATus:QUESTionable[:EVENT]?	9
STATus:QUESTionable:CONDition?.....	9
STATus:QUESTionable:ENABLE<Nrf>	9
STATus:QUESTionable:NTRansition	10
STATus:QUESTionable:PTRansition.....	10
STATus:OPERation[:EVENT]?	11
STATus:OPERation:CONDition?.....	11
STATus:OPERation:ENABLE.....	11
STATus:OPERation:NTRansition	12
STATus:OPERation:PTRansition.....	12
Chapter3 Current Measurement Commands.....	14
FETCh?	14
FETCh[:SCALar]:CURRent:AC?	14
MEASure[:SCALar]:CURRent:AC?	14
FETCh[:SCALar]:CURRent:CFActor?	14
MEASure[:SCALar]:CURRent:CFActor?	15
FETCh[:SCALar]:CURRent:DC?	15
MEASure[:SCALar]:CURRent:DC?	15
FETCh[:SCALar]:CURRent:RMS?	15
MEASure[:SCALar]:CURRent:RMS?.....	16
FETCh[:SCALar]:CURRent:RMN?.....	16
MEASure[:SCALar]:CURRent:RMN?.....	16
FETCh[:SCALar]:CURRent:MN?	16
MEASure[:SCALar]:CURRent:MN?	17
FETCh[:SCALar]:CURRent:INRush?	17
MEASure[:SCALar]:CURRent:INRush?	17
FETCh[:SCALar]:CURRent:MAXPk?	17
MEASure[:SCALar]:CURRent:MAXPk?	18
FETCh[:SCALar]:CURRent:MINPk?	18
MEASure[:SCALar]:CURRent:MINPk?	18
FETCh[:SCALar]:CURRent:PPEak?	18
MEASure[:SCALar]:CURRent:PPEak?	19
Chapter4 Voltage Measurement Commands.....	20
FETCh[:SCALar]:VOLTage:AC?	20
MEASure[:SCALar]:VOLTage:AC?	20
FETCh[:SCALar]:VOLTage:DC?	20

MEASure[:SCALar]:VOLTage:DC?	20
FETCh[:SCALar]:VOLTage:RMS?	21
MEASure[:SCALar]:VOLTage:RMS?	21
FETCh[:SCALar]:VOLTage:RMN?	21
MEASure[:SCALar]:VOLTage:RMN?	21
FETCh[:SCALar]:VOLTage:MN?	21
MEASure[:SCALar]:VOLTage:MN?	22
FETCh[:SCALar]:VOLTage:CFACtor?	22
MEASure[:SCALar]:VOLTage:CFACtor?	22
FETCh[:SCALar]:VOLTage:MAXPk?	22
MEASure[:SCALar]:VOLTage:MAXPk?	23
FETCh[:SCALar]:VOLTage:MINPk?	23
MEASure[:SCALar]:VOLTage:MINPk?	23
FETCh[:SCALar]:VOLTage:PPEak?	23
MEASure[:SCALar]:VOLTage:PPEak?	24
Chapter5 Power Measurement Commands	25
FETCh[:SCALar]:POWEr:ACTive?	25
MEASure[:SCALar]:POWEr:ACTive?	25
FETCh[:SCALar]:POWEr:APParent?	25
MEASure[:SCALar]:POWEr:APParent?	25
FETCh[:SCALar]:POWEr:REACTive?	26
MEASure[:SCALar]:POWEr:REACTive?	26
FETCh[:SCALar]:POWEr:PFACtor?	26
MEASure[:SCALar]:POWEr:PFACtor?	26
FETCh[:SCALar]:POWEr:PHASe?	26
MEASure[:SCALar]:POWEr:PHASe?	27
Chapter6 Frequency Measurement Commands	28
FETCh[:SCALar]:FREQuency:VOLTage?	28
MEASure[:SCALar]:FREQuency:VOLTage?	28
FETCh[:SCALar]:FREQuency:CURRent?	28
MEASure[:SCALar]:FREQuency:CURRent?	28
FETCh[:SCALar]:FREQuency:SSOURce?	29
MEASure[:SCALar]:FREQuency:SSOURce?	29
Chapter7 Commands	30
FETCh[:SCALar]:ENERgy[:ACTive][:SUM]?	30
MEASure[:SCALar]:ENERgy[:ACTive][:SUM]?	30
FETCh[:SCALar]:ENERgy[:ACTive]:POSitive?	30
MEASure[:SCALar]:ENERgy[:ACTive]:POSitive?	30
FETCh[:SCALar]:ENERgy[:ACTive]:NEGative?	31
MEASure[:SCALar]:ENERgy[:ACTive]:NEGative?	31
FETCh[:SCALar]:ENERgy:CHARge[:SUM]?	31
MEASure[:SCALar]:ENERgy:CHARge[:SUM]?	31
FETCh[:SCALar]:ENERgy:CHARge:POSitive?	32
MEASure[:SCALar]:ENERgy:CHARge:POSitive?	32
FETCh[:SCALar]:ENERgy:CHARge:NEGative?	32
MEASure[:SCALar]:ENERgy:CHARge:NEGative?	32
FETCh[:SCALar]:ENERgy:TIME?	32
MEASure[:SCALar]:ENERgy:TIME?	33
FETCh[:SCALar]:ENERgy[:ACTive]:AVERage?	33
MEASure[:SCALar]:ENERgy[:ACTive]:AVERage?	33
Chapter8 Harmonic Measurement Commands	34
FETCh[:SCALar]:HARMonics:CURRent:AMPLitude?	34
MEASure[:SCALar]:HARMonics:CURRent:AMPLitude?	34
FETCh[:SCALar]:HARMonics:CURRent:FUNDamental?	34
MEASure[:SCALar]:HARMonics:CURRent:FUNDamental?	35

FETCh[:SCALar]:HARMonics:CURRent:THARmonic?	35
MEASure[:SCALar]:HARMonics:CURRent:THARmonic?	35
FETCh[:SCALar]:HARMonics:CURRent:THDistort?	35
MEASure[:SCALar]:HARMonics:CURRent:THDistort?	36
FETCh[:SCALar]:HARMonics:VOLTagE:AMPLitude?	36
MEASure[:SCALar]:HARMonics:VOLTagE:AMPLitude?	36
FETCh[:SCALar]:HARMonics:VOLTagE:FUNDamental?	36
MEASure[:SCALar]:HARMonics:VOLTagE:FUNDamental?	37
FETCh[:SCALar]:HARMonics:VOLTagE:THARmonic?	37
MEASure[:SCALar]:HARMonics:VOLTagE:THARmonic?	37
FETCh[:SCALar]:HARMonics:VOLTagE:THDistort?	37
MEASure[:SCALar]:HARMonics:VOLTagE:THDistort?	38
FETCh[:SCALar]:HARMonics:POWEr[:ACTive]:AMPLitude?	38
MEASure[:SCALar]:HARMonics:POWEr[:ACTive]:AMPLitude?	38
FETCh[:SCALar]:HARMonics:POWEr[:ACTive]:FUNDamental?	39
MEASure[:SCALar]:HARMonics:POWEr[:ACTive]:FUNDamental?	39
FETCh[:SCALar]:HARMonics:POWEr[:ACTive]:THARmonic?	39
MEASure[:SCALar]:HARMonics:POWEr[:ACTive]:THARmonic?	39
FETCh[:SCALar]:HARMonics:POWEr[:ACTive]:THDistort?	40
MEASure[:SCALar]:HARMonics:POWEr[:ACTive]:THDistort?	40
FETCh[:SCALar]:HARMonics:POWEr:APParent?	40
MEASure[:SCALar]:HARMonics:POWEr:APParent?	40
FETCh[:SCALar]:HARMonics:POWEr:REACTive?	41
MEASure[:SCALar]:HARMonics:POWEr:REACTive?	41
FETCh[:SCALar]:HARMonics:POWEr:PFACtor?	41
MEASure[:SCALar]:HARMonics:POWEr:PFACtor?	42
FETCh[:SCALar]:HARMonics:POWEr:PHASe:UU?	42
MEASure[:SCALar]:HARMonics:POWEr:PHASe:UU?	42
FETCh[:SCALar]:HARMonics:POWEr:PHASe:UI?	43
MEASure[:SCALar]:HARMonics:POWEr:PHASe:UI?	43
FETCh[:SCALar]:HARMonics:POWEr:PHASe:II?	43
MEASure[:SCALar]:HARMonics:POWEr:PHASe:II?	43
Chapter9 Sense Relevant Commands	45
[:SENSe]AVERAge[:STATe] <bool>	45
[:SENSe]AVERAge:TCONtrol <name>	45
[:SENSe]AVERAge:TYPE <boolean>	45
[:SENSe]AVERAge:COUNT <n>	45
[:SENSe]CURRent:RANGe:AUTO <bool>	46
[:SENSe]VOLTagE:RANGe:AUTO <bool>	46
[:SENSe]CURRent:RANGe <NRf>	46
[:SENSe]VOLTagE:RANGe <NRf>	47
[:SENSe]CURRent:EXS1[:STATe] 	47
[:SENSe]CURRent:EXS2[:STATe] 	47
[:SENSe]CURRent:SRATio:EXS1 <NRf>	48
[:SENSe]CURRent:SRATio:EXS2 <NRf>	48
[SENSe:]CURRent:EXS:RANGe	48
Chapter10 Trigger Commands	50
ABORT	50
INITiate:CONTInuous <state>	50
INITiate[:IMMediate]	50
TRIGGer:IMMediate	50
TRIGGer:SOURce <state>	51
TRIGGer:SLOPe <state>	51
TRIGGer:VOLTagE:LEVel <level>	51
TRIGGer:CURRent:LEVel <level>	52
Chapter11 Oscilloscope Commands	53

WAVE:TRIG:SOUR< VOLTage CURRent EXTernal >	53
WAVE:TRIG:SLOP <POSitive NEGative ANY>	53
WAVE:TRIG:MODE <AUTO NORMAl>	53
WAVE:TRIG:DELAy:TIME <n>	53
WAVE:TRIG:DIVTime (UNIT:S) <n>	54
WAVE:RUN	54
WAVE:STOP	54
WAVE:SINGLE	55
WAVE:VOLTage:DATA[:NORMAlization]?	55
WAVE:CURRent:DATA[:NORMAlization]?	55
WAVE:TRIGger[:STATe]?	55
Chapter12 INPut Commands.....	57
[:INPut]:HARMOnics:ORDer <NRI>	57
[:INPut]:HARMonic:PLLSourCe <OFF U I>	57
[:INPut]:HARMOnics:THD <THDR THDF>	57
[:INPut]:HARMonic:SEQuence <ALL ODD EVEN>	57
[:INPut]:SSourCe <OFF U I>	58
[:INPut]:CFACtor <NRI>	58
[:INPut]:FILTer:FREQUency <bool>	58
[:INPut]:FILTer:LINE <bool>	59
[:INPut]:RATE <NRI>	59
[INPut]::INTEgral:ACAL <bool>	59
[INPut]:INTEgral:QMODE {RMS MN DC RMN AC}	59
[INPut]:INTEgral:WPTyPe {CHARge SOLD DISChArge BOUgHt }	60
[INPut:]INRush[:STATe] <OFF ON>	60
[INPut:]INRush:TRIGger:CURRent[:LEVel] <n>	60
[INPut:]INRush:DELAy:TIME <n>	61
[INPut:]INRush:MEASure:TIME <n>	61
Chapter13 CALCulate Commands.....	62
[CALCulate]:INTEgral[:STATe]	62
[CALCulate]:INTEgral:CLEar[:IMMEdiate]	62
[CALCulate]:INTEgral:CLEar:AUTO <ON OFF>	62
[CALCulate]:INTEgral:STARt:SOURce < TIME MAN>	62
[:CALCulate]:INTEgral:STARt[:IMMEdiate]	63
[:CALCulate]:INTEgral:STOP:SOURce< TIME MAN TINTErval>	63
[:CALCulate]:INTEgral:STOP[:IMMEdiate]	64
[:CALCulate]:INTEgral:STOP:TINTErval < hhhh,mm,ss >	64
INTEgral:CONDition?	65
[:CALCulate][:METer]:MAXHold[:STATe] <bool>	65
[:CALCulate][:METer]:CLEar[:IMMEdiate]	65
[:CALCulate]:HARMonic[:STATe] <bool>	65
[:CALCulate]:SCOPE[:STATe] <bool>	66
[CALCulate:]HOLD[:STATe] <bool>	66
Chapter14 SYSTem Commands	67
SYSTem:BEEPer:IMMEdiate	67
SYSTem:BEEPer[:STATe] <bool>	67
SYSTem:DATE <NRf>,<NRf>,<NRf>	67
SYSTem:TIME <NRf>,<NRf>,<NRf>	67
SYSTem:KEY <NRI>	68
SYSTem:ERRor?	68
SYSTem:LANGuage <NRI>	68
SYSTem:LOCAl	68
SYSTem:REMote	69
SYSTem:CLEar	69
SYSTem:RWLock	69
SYSTem:VERSion?	69

Chapter15	Calibration Commands	71
	CALibration:ZERO	71
	CALibrate:SECure[:STATE]	71
	CALibrate:INITial	71
	CALibrate:SAVe	72
	CALibrate:CURRent:POINt	72
	CALibrate:CURRent:EXTernal:POINt<point>	73
	CALibrate:CURRent:EXTernal:[LEVel] <NRf>	73
	CALibrate:VOLTage:POINt	73
	CALibrate:VOLTage[:LEVel]	74
Chapter16	IEEE-488 Commands	75
	*CLS	75
	*ESE	75
	*ESR?	75
	*IDN?	76
	*OPC	76
	*RST	77
	*SRE	77
	*STB?	77
	*WAI	78

Chapter1 Remote Control

1.1 Overview

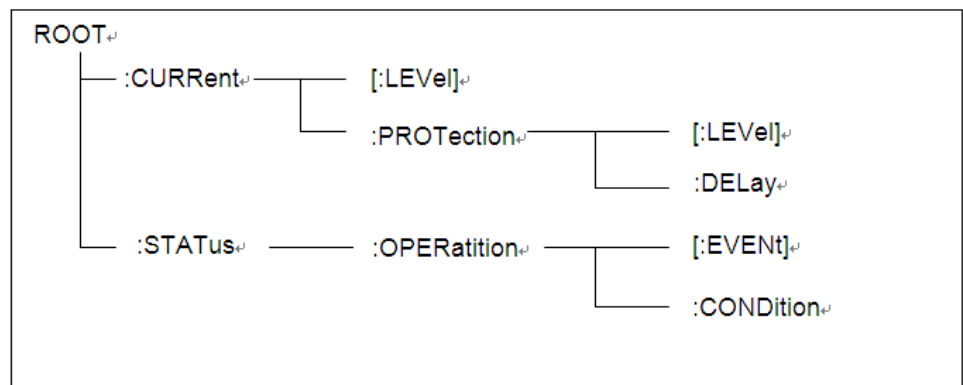
This chapter will provide following remote configuration introductions:

- Command format
- Data format
- Remote Operation

1.2 Command type

Two command types for SCPI: common command and subsystem

- Common command is basically irrelevant to specific operation, however, it controls overall functionality of the instrument, such as reset, status and synchronization. All common commands are trilateral commands marked with asterisk: *RST *IDN?*SRE.
- The subsystem command executes specified instrument functions. They are organized into a reverse tree structure with root on the top. A part of subsystem command tree is shown in the following chart, with which you can obtain commands from different paths. See the following chart for part command tree.



Multiple commands in a message

Several SCPI commands can be merged and sent as a single message with one information terminator.

Two aspects shall be considered when sending several commands in a single message:

- Commands in a message are separated by a semicolon.
- 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

Example:

```
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 in different subsystems, it is required to set the head path of message as a null string. If a command is started by a colon, the operation will discard all current head paths. For example, you may apply the following root specification to remove the output protection and inspect the status of operation status register in a message.

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

The following command indicates how to combine commands from different subsystems as if in the same subsystem:

```
POWEr:LEVEl 200;PROTection 28; :CURRent:LEVEl 3;PROTection:STATe ON
```

Note to use the optional head LEVEl to maintain path in the voltage and current subsystems and use the root specification to move among subsystems.

Including common command

You can combine the common command and subsystem command in the same message and consider the common command as a message unit, and separate it by a semicolon (separator of message unit). Common commands will not influence the head path; you can insert them into any place of the message.

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

```
OUTPut OFF;*RCL 2;OUTPut ON
```

```
IT872X-3X SCPI Communication Protocol
```

```
17
```

Case sensitivity

Common commands and SCPI commands are case insensitive: you may use upper or lower case or both, such as:

```
*RST = *rst
```

```
:DATA? = :data?
```

```
:SYSTem:PRESet = :system:preset
```

Long or short format

A SCPI command word can be sent in short or long format. The command subsystem table in the Chapter 5 provides long-format words. However, short command words are expressed in uppercase characters:

```
:SYSTem:PRESet long format
```

```
:SYST:PRES short format
```

```
:SYSTem:PRES combination of long and short formats
```

Note that every command word must only be long or short format, and the long and short command words cannot be interleaved. For example: :SYSTe:PRESe is an illegal command word which will cause an error. This command will not be executed.

Check

Observe the following query alert:

- To set a proper number of variables for the returned data, for example, if you are reading a measurement sequence, you must provide fractal dimension to the sequence according to measurement number stored in the measurement cache.
- Read back all query results before sending any command to the instrument.

Otherwise an error of query interrupt will occur and missing data will not be sent back.

1.3 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 parameters) 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 parameters) 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 parameters, 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 parameters. 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 parameters, then a comma must be used to separate adjacent parameters.

Space

You must use blank characters, [TAB] or [Space] to separate parameters 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 parameters. 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.

1.4 Data type

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

- Numerical parameter
- Commands requiring numerical parameters support the notations of all common decimal notations, including optional signs, decimal points, scientific notation, etc. Special values of numerical parameters are also

acceptable, such as MIN, MAX and DEF. In addition, suffixes for engineering units can also be sent together with numerical parameters (including M, k, m or u). If the command accepts only some specific values, the instrument will automatically round the input parameters to acceptable values. The following commands require numerical parameters of frequency value:

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

- Discrete parameter
- Discrete parameters 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 parameters in short format. The following commands require discrete parameters in voltage unit:

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

- Boolean parameter
- Boolean parameters 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 parameters are required by the following commands:

DISPlay {OFF|0|ON|1}

- ASCII string parameters
- String parameters 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...'

1.5 Remote Operation

There are four types of communication interfaces available:USB,Ethernet,GPIB and RS232.You can choose any one of them to communicate with a PC.

RS-232 Interface

RS232 interface:use a cable with two COM interface (DB9) to connect power meter and PC. It can be activated by menu key on the front panel.All SCPI commands are available through RS-232 programming.

- RS-232 data format
The RS-232 data is a 10-bit word with one start bit and one stop bit. The number of start and stop bits is not programmable.
- Baudrate
Available baudrate:4800 9600 19200 38400 57600 115200
- RS-232 connection
The RS-232 serial port can be connected to the serial port of a controller (i.e., personal computer) using a straight through RS-232 cable terminated with DB-9 connectors. Do not use a null modem cable. Table 2—2 shows the pinout for the connector.

If your computer uses a DB-25connector for the RS-232interface, you will

need a cable or adapter with a DB-25 connector on one end and a DB-9 connector on the other, wired straight through(not null modem).



Base pin number	Description
1	No conjunction
2	TXD, data transmission
3	RXD, data receiving
4	No conjunction
5	GND, grounding
6	No conjunction
7	CTS, clear to send
8	RTS, request to send
9	No conjunction

RS-232 troubleshooting:

If you are having trouble communicating over the RS-232 interface, check the following:

The computer and the power meter must be configured for the same baudrate, parity, number of data bits, and flow control options. Note that the electronic load is configured for 1 start bit and 1 stop bit (these values are fixed).

The correct interface cables or adapters must be used, as described under RS-232 connector. Note that even if the cable has the proper connectors for your system, the internal wiring may be incorrect.

The interface cable must be connected to the correct serial port on your computer (COM1, COM2, etc.).

Communication setting

Before communication operation, please make sure that the following parameters of electronic load match that of PC.

Baud rate : 9600 (4800, 9600, 19200, 38400, 57600, 115200). You can enter system menu through panel to set communication baud rate.

Data bit: 8 Stop bit: 1 Parity: (none, even, odd)

EVEN: : eight data bits with even parity

ODD: : eight data bits with odd parity

NONE: : eight data bits without parity

Start Bit	8 Data Bits	Parity=None	Stop Bit
-----------	-------------	-------------	----------

USB interface

Use Type A to Type B USB cables to connect the power meter.

The USB interface capabilities of the power meter are described below:

- The interface is IEEE488.2 standard USB488 interface
- The interface accepts REN_CONTROL, GO_TO_LOCAL, LOCAL_LOCKOUT request.
- The interface accepts MsgID = TRIGGER USBTMC command message and forwards TRIGGER requests to the function layer.

The USB488 device capabilities of the power meter are described below:

- The device understands all mandatory SCPI commands.
- The device is SR1 capable.
- The device is RL1 capable.

- The device is DT1 capable.

GPIB interface

First connect GPIB port of power meter to GPIB card of PC. They must be sufficient contact and tighten the screws. And then set address. The address can be set from 0 to 30.

LAN interface

Use a network cable to connect PC through LAN interface of the power meter.

- IP mode
MANU: manually set IP address, subnet mask, default gateway, etc.
DHCP (Dynamic Host Configuration Protocol)
DHCP refers to a protocol that temporarily assigns necessary information to PCs connected to the Internet. If the network is provided with an available DHCP server, the server will automatically assign information (IP address, subnet mask and default gateway) to PCs and other equipment connected to the network. IT9120 series cannot use DHCP unless the network is provided with a DHCP server. Please ask your network administrator DHCP if DHCP is available.
- IP Address
IP addresses assigned to IT9120 series can be set. Default address is 192.168.000.000.
 - ◆ IP address refers to ID assigned by the network to each equipment (Internet or enterprise Intranet).
 - ◆ IP address refers to four 32-bit values within 0 - 255 which are separated by decimal points, such as [192.168.111.24].
 - ◆ Please ask the network administrator to provide an IP address.
 - ◆ Use DHCP network to automatically set the IP address.
- Subnet masksetting
The mask used when subnet website is determined from an IP address may be set. Default address is 255.255.255.255.
 - ◆ Vast TCP/IP networks like Internet are often divided into several small networks, i.e. subnets. The subnet mask is used to identify how many digits of the 32-digit values in the IP address belong to the subnet address. The part beyond the network address is used to identify the host number of each PC connected to the network.
 - ◆ Ask your network administrator about values of subnet mask.
 - ◆ Use DHCP network to automatically set the subnet mask.
- Gatewaysetting
IP address of gateway may be set to communicate with equipment in other network (default gateway). Default address is 192.168.000.000.
 - ◆ The default gateway controls data exchange between networks and protocols to ensure smooth data transmission.
 - ◆ Ask your network administrator about values of default gateway. It is possible that the setting is unnecessary.
Use DHCP network to automatically set the default gateway.

Chapter2 Status Register Commands

IT9120 series has two kinds of status register: Operation status register and Query status register. Using status register's commands can set or query status register.

Standard Event Register

Event register is a kind of read-only memory. It is used to save the current execution situation of power meter. Event register's data applies latch format, once the data is saved, the following data will be ignored. Sending command *RST or restart the power meter will not change the data in event register. But if query the event register or sending clear command *CLS, the data will be cleared automatically.

The standard event register includes following status: output status, command syntax error, command execution error, selftest or calibration error, query error etc.

Bit	Definition
0	OPC Operation Complete. Set to 1 when the operation designated by the *OPC command has been completed.
1	RQC Not used ,Always set to 0.
2	QYE Query Error. Set to 1 if the output queue is empty or if the data is missing even after a query has been sent.
3	DDE Device Error. Set to 1 when execution of the command is not possible due to an internal problem in the Instrument that is not a command error or an execution error.
4	EXE Execution Error. Set to 1 when the command syntax is correct but the command cannot be executed in the current state.
5	CME Command Error. Set to 1 when the command syntax is incorrect.
6	Not Used Not Used ,Always set to 0.
7	PON Power On. Set to 1 when the power is turned ON.

Status Byte Register

Status byte register is used to record other register's message. Among which the query data is temporarily stored into the output buffer of power meter and feedback to the operator via bit4. Status byte register's data will not be latched. Once the event register's status changed, then the corresponding bit of status byte register will be changed too.

Bit	Definition
0	MSB Measurement Summary Bit
1	Not Used Always set to 0.
2	EAV Error Available
3	QSB Questionable Summary Bit
4	MAV Message Available
5	ESB Event Summary Bit
6	RQS Request for service
7	OSB OSB(Operation Summary Bit)

STATus:QUEStionable[:EVENT]?

This command is used to read the value of query event register. Power meter will return a decimal number which is the binary weighted sum of each bits of register. All these bits value is latched and will be cleared after executing this command.

Query syntax

STATus:QUEStionable[:EVENT]?

Parameters

None

Returned value

<NR1>

Related commands

STATus:QUEStionable:ENABle

STATus:QUEStionable:CONDition?

This command is used to read the value of query condition register to get the power meter's status, such as OCPeak/OCRMS/OV/OP/OT.

Query syntax

STATus:QUEStionable:CONDition?

Parameters

None

Returned value

<NR1>

STATus:QUEStionable:ENABle<NRf>

This command is used to set the value of query event enable registers. This register is a mask of enabling specific bits from the questionable event register to set the questionable summary bit(ques) of the status byte register.

Command syntax:

STATus:QUEStionable:ENABle <NRf>

Parameters:

0~65535

Power-on value

Refer to *PSC command

Example

STATus:QUEStionable:ENABle 16

Query syntax

STATus:QUEStionable:ENABLE?

Returned value

<NR1>

Related command

*PSC

Bit 0	Voltage summary questionable
Bit 1	Current summary questionable
Bit 5	Frequency questionable
Bit 7	Sync questionable (lost sync)
Bit 8	PLL source questionable (lost pll)

STATus:QUEStionable:NTRansition

This command set the value of the Questionable NTR(Negative-Transition) register. The register serve as polarity filters between the Questionable Condition and Questionable Event registers to cause the following actions. When a bit of Questionable NTR register is set to 1, then a 1-to-0 transition of the corresponding bit of the Questionable Condition register causes that bit in the Questionable Event register to be set.

Command syntax

STATus:QUEStionable:NTRansition <NR1>

Parameters

0~65535

Example

STATus:QUEStionable:NTRansition 128

Query syntax

STATus:QUEStionable:NTRansition?

STATus: QUEStionable:PTRansition

This command set the value of the Questionable PTR(Positive-Transition) register. The register serve as polarity filters between the Questionable Condition and Questionable Event registers to cause the following actions. When a bit of Questionable PTR register is set to 1, then a 1-to-0 transition of the corresponding bit of the Questionable Condition register causes that bit in the Questionable Event register to be set.

Command syntax

STATus:QUEStionable:PTRansition <NR1>

Parameters:

0~65535

Example

STATus:QUEStionable:PTRansition 128

Query syntax

STATus:QUEStionable:PTRansition?

STATus:OPERation[:EVENT]?

This query returns the value of the Operation Event register. The Event register is a read-only register, which stores (latches) all events that are passed by the Operation NTR and/or PTR filter. Reading the Operation Event register clears it.

Query syntax

STATus:OPERation[:EVENT]?

Parameters

None

Returned values

<NR1>

Related command

STATus:OPERation:ENABLE

The bit configuration of the Operation status registers is as follows:

Bit 2	Ranging
Bit 3	ITG(Integrate Busy)
Bit 4	ITM(Integrate Timer Busy)
Bit 5	Waiting for trigger

STATus:OPERation:CONDition?

This query returns the value of the Operation Condition register. That is a read-only register, which holds the live (unlatched) operational status of the power meter.

Query syntax:

STATus:OPERation:CONDition?

Parameters:

None

Returned value:

<NR1>

STATus:OPERation:ENABLE

This command set the value of the Operational 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.

Command syntax:

STATus:OPERation:ENABLE <NR1>

Parameters:

0~65535

Example:

STATus:OPERation:ENABLE 128

Query syntax:

STATus: OPERation:ENABLE?

Returned value:

<NR1>

STATus:OPERation:NTRansition

This command set the value of the Operation NTR(Negative-Transition) register. The programmed parameters define the specific bits from operation event register which will cause the operation summary bit(OPER) of the status register to be 1.

Command syntax:

STATus:OPERation:NTRansition <NR1>

Parameters:

0~65535

Example:

STATus:OPERation:NTRansition 128

Query syntax:

STATus:OPERation:NTRansition

STATus:OPERation:PTRansition

This command set the value of the Operation PTR(Positive-Transition) register. The programmed parameters define the specific bits from operation event register which will cause the operation summary bit(OPER) of the status register to be 1.

Command syntax:

STATus:OPERation:PTRansition <NR1>

Parameters

0~65535

Example

STATus:OPERation:PTRansition 128

Chapter3 Current Measurement Commands

This chapter describes the current relevant commands.

FETCH?

This command is used to read all relevant parameters of current, voltage and power measured by the power meter. The order in which the parameters are read is as follow.

VOLT_RMS, VOLT_MN, VOLT_RMN, VOLT_DC, VOLT_AC, VOLT_MAXPk, VOLT_MINPk, VOLT_PPEak, VOLT_CF, FREQ_VOLT, CURR_RMS, CURR_MN, CURR_RMN, CURR_DC, CURR_AC, CURR_MAXPk, CURR_MINPk, CURR_PPEak, CURR_CF, FREQ_CURR, CURR_INR, POWER_Active, POWER_REActive, POWER_APParent, POWER_PF, POWER_Phase, FREQ_SSource

Please refer to IT9120 User Manual for the detailed instructions of the parameters.

Command syntax:

FETCH?

Returned value

<NR2>, <NR2>,, <NR2>

FETCH[:SCALar]:CURRent:AC?

This command is used to read the latest pre-processing current. It will not affect the device setting values and will not trigger a new measurement operation. It only requires to get the latest available value that has been measured.

Command syntax:

FETCH[:SCALar]:CURRent:AC?

Returned value

<NR2>

MEASure[:SCALar]:CURRent:AC?

This command will trigger the measurement operation to get a real-time current.

Command syntax:

MEASure[:SCALar]:CURRent:AC?

Returned value:

<NR2>

FETCH[:SCALar]:CURRent:CFACTOR?

This command is used to read the latest pre-processing current crest factor. It

will not affect the device setting values and will not trigger a new measurement operation. It only requires to get the latest available value that has been measured.

Command syntax:

FETCh[:SCALar]:CURRent:CFACtor?

Returned value

<NR2>

MEASure[:SCALar]:CURRent:CFACtor?

This command will trigger the measurement operation to get a real-time current crest factor.

Command syntax:

MEASure[:SCALar]:CURRent:CFACtor?

Returned value

<NR2>

FETCh[:SCALar:]CURRent:DC?

This command is used to read the latest pre-processing average current. It will not affect the device setting values and will not trigger a new measurement operation. It only requires to get the latest available value that has been measured.

Command syntax:

FETCh[:SCALar:]CURRent:DC?

Returned value

<NR2>

MEASure[:SCALar:]CURRent:DC?

This command will trigger the measurement operation to get a real-time average current.

Command Syntax

MEASure[:SCALar:]CURRent:DC?

Returned value

<NR2>

FETCh[:SCALar:]CURRent:RMS?

This command is used to read the latest pre-processing current effective value. It will not affect the device setting values and will not trigger a new measurement operation. It only requires to get the latest available value that has

been measured.

Command Syntax:

FETCh[:SCALar:]CURRent:RMS?

Returned value

<NR2>

MEASure[:SCALar:]CURRent:RMS?

This command will trigger the measurement operation to get a real-time current effective value.

Command Syntax:

MEASure[:SCALar:]CURRent:RMS?

Returned value:

<NR2>

FETCh[:SCALar:]CURRent:RMN?

This command is used to read the latest pre-processing current rectified mean value(RMN).It will not affect the device setting values and will not trigger a new measurement operation.It only requires to get the latest available value that has been measured.

Command Syntax

FETCh[:SCALar:]CURRent:RMN?

Returned value

<NR2>

MEASure[:SCALar:]CURRent:RMN?

This command will trigger the measurement operation to get a real-time current rectified mean value(RMN).

Command Syntax:

MEASure[:SCALar:]CURRent:RMN?

Returned value:

<NR2>

FETCh[:SCALar:]CURRent:MN?

This command is used to read the latest pre-processing MN(Calibration to the average rectified value of the effective value) of current.It will not affect the device setting values and will not trigger the measurement operation.It only requires to get the latest value that has been measured.

Command Syntax:

FETCh[:SCALar:]CURRent:MN?

Returned value:

<NR2>

MEASure[:SCALar:]CURRent:MN?

This command is used to read real-time MN(Calibration to the average rectified value of the effective value) of current.

Command Syntax

MEASure[:SCALar:]CURRent:MN?

Returned value

<NR2>

FETCh[:SCALar:]CURRent:INRush?

This command is used to read the latest pre-processing inrush current. It will not affect the device setting values and will not trigger a new measurement operation. It only requires to get the latest available value that has been measured.

Command Syntax:

FETCh[:SCALar:]CURRent:INRush?

Returned value

<NR2>

MEASure[:SCALar:]CURRent:INRush?

This command will trigger the measurement operation to get a real-time inrush current.

Command Syntax

MEASure[:SCALar:]CURRent:INRush?

Returned value

<NR2>

FETCh[:SCALar:]CURRent:MAXPk?

This command is used to read the latest pre-processing positive peak current. It will not affect the device setting values and will not trigger a new measurement operation. It only requires to get the latest available value that has been measured.

Command Syntax:

FETCh[:SCALar:]CURRent:MAXPk?

Returned value

<NR2>

MEASure[:SCALar]:CURRent:MAXPk?

This command will trigger the measurement operation to get a real-time positive peak current

Command Syntax:

MEASure[:SCALar]:CURRent:MAXPk?

Returned value

<NR2>

FETCh[:SCALar]:CURRent:MINPk?

This command is used to read the latest pre-processing negative peak current. It will not affect the device setting values and will not trigger a new measurement operation. It only requires to get the latest available value that has been measured.

Command Syntax

FETCh[:SCALar]:CURRent:MINPk?

Returned value

<NR2>

MEASure[:SCALar]:CURRent:MINPk?

This command will trigger the measurement operation to get a real-time negative peak current

Command Syntax

MEASure[:SCALar]:CURRent:MINPk?

Returned value

<NR2>

FETCh[:SCALar]:CURRent:PPEak?

This command is used to read the latest pre-processing peak-peak current. It will not affect the device setting values and will not trigger a new measurement operation. It only requires to get the latest available value that has been measured.

Command Syntax

FETCh[:SCALar]:CURRent:PPEak?

Returned value

<NR2>

MEASure[:SCALar]:CURRent:PPEak?

This command will trigger the measurement operation to get a real-time peak-peak current.

Command Syntax

MEASure[:SCALar]:CURRent:PPEak?

Returned value

<NR2>

Chapter4 Voltage Measurement Commands

FETCh[:SCALar]:VOLTage:AC?

This command is used to read the latest pre-processing AC voltage. It will not affect the device setting values and will not trigger a new measurement operation. It only requires to get the latest available value that has been measured.

Command Syntax

FETCh[:SCALar]:VOLTage:AC?

Returned Value

<NR2>

MEASure[:SCALar]:VOLTage:AC?

This command will trigger the measurement operation to get a real-time AC voltage.

Command Syntax

MEASure[:SCALar]:VOLTage:AC?

Returned value

<NR2>

FETCh[:SCALar]:VOLTage:DC?

This command is used to read the latest pre-processing DC average voltage. It will not affect the device setting values and will not trigger a new measurement operation. It only requires to get the latest available value that has been measured.

Command Syntax

FETCh[:SCALar]:VOLTage:DC?

Returned value

<NR2>

MEASure[:SCALar]:VOLTage:DC?

This command will trigger the measurement operation to get a real-time DC average voltage.

Command Syntax:

MEASure[:SCALar]:VOLTage:DC?

Returned value

<NR2>

FETCh[:SCALar]:VOLTage:RMS?

This command is used to read the latest pre-processing voltage effective value. It will not affect the device setting values and will not trigger a new measurement operation. It only requires to get the latest available value that has been measured.

Command Syntax

```
FETCh[:SCALar]:VOLTage:RMS?
```

Returned value

```
<NR2>
```

MEASure[:SCALar]:VOLTage:RMS?

This command will trigger the measurement operation to get a real-time voltage effective value.

Command Syntax

```
MEASure[:SCALar]:VOLTage:RMS?
```

Returned value

```
<NR2>
```

FETCh[:SCALar]:VOLTage:RMN?

This command is used to read the latest pre-processing voltage rectified mean value. It will not affect the device setting values and will not trigger a new measurement operation. It only requires to get the latest available value that has been measured.

Command Syntax

```
FETCh[:SCALar]:VOLTage:RMN?
```

Returned value

```
<NR2>
```

MEASure[:SCALar]:VOLTage:RMN?

This command will trigger the measurement operation to get a real-time voltage rectified mean value.

Command Syntax

```
MEASure[:SCALar]:VOLTage:RMN?
```

Returned value

```
<NR2>
```

FETCh[:SCALar]:VOLTage:MN?

This command is used to read the latest pre-processing MN(Calibration to the average rectified value of the effective value) of voltage. It will not affect the

device setting values and will not trigger the measurement operation. It only requires to get the latest value that has been measured.

Command Syntax

FETCh[:SCALar]:VOLTage:MN?

Returned value

<NR2>

MEASure[:SCALar]:VOLTage:MN?

This command is used to read real-time MN (Calibration to the average rectified value of the effective value) of voltage.

Command Syntax

MEASure[:SCALar]:VOLTage:MN?

Returned value

<NR2>

FETCh[:SCALar]:VOLTage:CFACTor?

This command is used to read the latest pre-processing voltage crest factor. It will not affect the device setting values and will not trigger a new measurement operation. It only requires to get the latest available value that has been measured.

Command Syntax

FETCh[:SCALar]:VOLTage:CFACTor?

Returned value

<NR2>

MEASure[:SCALar]:VOLTage:CFACTor?

This command will trigger the measurement operation to get a real-time voltage crest factor.

Command Syntax

MEASure[:SCALar]:VOLTage:CFACTor?

Returned value

<NR2>

FETCh[:SCALar]:VOLTage:MAXPk?

This command is used to read the latest pre-processing positive peak voltage. It will not affect the device setting values and will not trigger a new measurement operation. It only requires to get the latest available value that has been measured.

Command syntax

FETCh[:SCALar]:VOLTage:MAXPk?

Returned value

<NR2>

MEASure[:SCALar]:VOLTage:MAXPk?

This command will trigger the measurement operation to get a real-time positive peak voltage.

Command Syntax

MEASure[:SCALar]:VOLTage:MAXPk?

Returned value

<NR2>

FETCh[:SCALar]:VOLTage:MINPk?

This command is used to read the latest pre-processing negative peak voltage. It will not affect the device setting values and will not trigger a new measurement operation. It only requires to get the latest available value that has been measured.

Command Syntax

FETCh[:SCALar]:VOLTage:MINPk?

Returned value

<NR2>

MEASure[:SCALar]:VOLTage:MINPk?

This command will trigger the measurement operation to get a real-time negative peak voltage.

Command Syntax

MEASure[:SCALar]:VOLTage:MINPk?

Returned value

<NR2>

FETCh[:SCALar]:VOLTage:PPEak?

This command is used to read the latest pre-processing peak-peak voltage. It will not affect the device setting values and will not trigger a new measurement operation. It only requires to get the latest available value that has been measured.

Command Syntax

FETCh[:SCALar]:VOLTage:PPEak?

Returned value

<NR2>

MEASure[:SCALar]:VOLTage:PPEak?

This command will trigger the measurement operation to get a real-time peak-peak voltage.

Command Syntax

MEASure[:SCALar]:VOLTage:PPEak?

Returned value

<NR2>

Chapter5 Power Measurement Commands

FETCh[:SCALar]:POWer:ACTive?

This command is used to read the latest pre-processing active power. It will not affect the device setting values and will not trigger a new measurement operation. It only requires to get the latest available value that has been measured.

Command Syntax

```
FETCh[:SCALar]:POWer:ACTive?
```

Returned value

```
<NR2>
```

MEASure[:SCALar]:POWer:ACTive?

This command will trigger the measurement operation to get a real-time active power.

Command Syntax

```
MEASure[:SCALar]:POWer:ACTive?
```

Returned value

```
<NR2>
```

FETCh[:SCALar]:POWer:APParent?

This command is used to read the latest pre-processing apparent power. It will not affect the device setting values and will not trigger a new measurement operation. It only requires to get the latest available value that has been measured.

Command Syntax

```
FETCh[:SCALar]:POWer:APParent?
```

Returned value

```
<NR2>
```

MEASure[:SCALar]:POWer:APParent?

This command will trigger the measurement operation to get a real-time apparent power.

Command Syntax

```
MEASure[:SCALar]:POWer:APParent?
```

Returned value

```
<NR2>
```

FETCh[:SCALar]:POWer:REACtive?

This command is used to read the latest pre-processing reactive power. It will not affect the device setting values and will not trigger a new measurement operation. It only requires to get the latest available value that has been measured.

Command Syntax

```
FETCh[:SCALar]:POWer:REACtive?
```

Returned value

```
<NR2>
```

MEASure[:SCALar]:POWer:REACtive?

This command will trigger the measurement operation to get a real-time reactive power.

Command Syntax

```
MEASure[:SCALar]:POWer:REACtive?
```

Returned value

```
<NR2>
```

FETCh[:SCALar]:POWer:PFACTOR?

This command is used to read the latest pre-processing power factor. It will not affect the device setting values and will not trigger a new measurement operation. It only requires to get the latest available value that has been measured.

Command Syntax

```
FETCh[:SCALar]:POWer:PFACTOR?
```

Returned value

```
<NR2>
```

MEASure[:SCALar]:POWer:PFACTOR?

This command will trigger the measurement operation to get a real-time power factor.

Command Syntax

```
MEASure[:SCALar]:POWer:PFACTOR?
```

Returned value

```
<NR2>
```

FETCh[:SCALar]:POWer:PHASe?

This command is used to read the latest pre-processing phase difference between voltage and current. It will not affect the device setting values and will

not trigger a new measurement operation. It only requires to get the latest available value that has been measured.

Command Syntax

FETCh[:SCALar]:POWER:PHASe?

Returned value

<NR2>

MEASure[:SCALar]:POWER:PHASe?

This command will trigger the measurement operation to get a real-time phase difference between voltage and current.

Command Syntax

MEASure[:SCALar]:POWER:PHASe?

Returned value

<NR2>

Chapter6 Frequency Measurement Commands

FETCh[:SCALar]:FREQUency:VOLTage?

This command is used to read the latest pre-processing voltage frequency. It will not affect the device setting values and will not trigger a new measurement operation. It only requires to get the latest available value that has been measured.

Command Syntax

```
FETCh[:SCALar]:FREQUency:VOLTage?
```

Returned value

```
<NR2>
```

MEASure[:SCALar]:FREQUency:VOLTage?

This command will trigger the measurement operation to get a real-time voltage frequency.

Command Syntax

```
MEASure[:SCALar]:FREQUency:VOLTage?
```

Returned value

```
<NR2>
```

FETCh[:SCALar]:FREQUency:CURRent?

This command is used to read the latest pre-processing current frequency. It will not affect the device setting values and will not trigger a new measurement operation. It only requires to get the latest available value that has been measured.

Command Syntax

```
FETCh[:SCALar]:FREQUency:CURRent?
```

Returned value

```
<NR2>
```

MEASure[:SCALar]:FREQUency:CURRent?

This command will trigger the measurement operation to get a real-time current frequency.

Command Syntax

```
MEASure[:SCALar]:FREQUency:CURRent?
```

Returned value

<NR2>

FETCh[:SCALAr]:FREQuency:SSOurce?

This command is used to read the latest pre-processing synchronous source frequency. It will not affect the device setting values and will not trigger a new measurement operation. It only requires to get the latest available value that has been measured.

Command Syntax

FETCh[:SCALAr]:FREQuency:SSOurce?

Returned value

<NR2>

MEASure[:SCALAr]:FREQuency:SSOurce?

This command will trigger the measurement operation to get a real-time synchronous source frequency.

Command Syntax

MEASure[:SCALAr]:FREQuency:SSOurce?

Returned value

<NR2>

Chapter7 Commands

FETCh[:SCALar]:ENERgy[:ACTive][:SUM]?

This command is used to read the latest pre-processing the sum of positive watt hour and negative watt hour. It will not affect the device setting values and will not trigger a new measurement operation. It only requires to get the latest available value that has been measured.

Command Syntax

FETCh[:SCALar]:ENERgy[:ACTive][:SUM]?

Returned value

<NR2>

MEASure[:SCALar]:ENERgy[:ACTive][:SUM]?

This command will trigger the measurement operation to get a real-time sum of positive watt hour and negative watt hour.

Command Syntax

MEASure[:SCALar]:ENERgy[:ACTive][:SUM]?

Returned value

<NR2>

FETCh[:SCALar]:ENERgy[:ACTive]:POSitive?

This command is used to read the latest pre-processing positive watt hour. It will not affect the device setting values and will not trigger a new measurement operation. It only requires to get the latest available value that has been measured.

Command Syntax

FETCh[:SCALar]:ENERgy[:ACTive]:POSitive?

Returned value

<NR2>

MEASure[:SCALar]:ENERgy[:ACTive]:POSitive?

This command will trigger the measurement operation to get a real-time positive watt hour.

Command Syntax

MEASure[:SCALar]:ENERgy[:ACTive]:POSitive?

Returned value

<NR2>

FETCh[:SCALar]:ENERgy[:ACTive]:NEGative?

This command is used to read the latest pre-processing negative watt hour. It will not affect the device setting values and will not trigger a new measurement operation. It only requires to get the latest available value that has been measured.

Command Syntax

FETCh[:SCALar]:ENERgy[:ACTive]:NEGative?

Returned value

<NR2>

MEASure[:SCALar]:ENERgy[:ACTive]:NEGative?

This command will trigger the measurement operation to get a real-time negative watt hour.

Command Syntax

MEASure[:SCALar]:ENERgy[:ACTive]:NEGative?

Returned value

<NR2>

FETCh[:SCALar]:ENERgy:CHARge[:SUM]?

This command is used to read the latest pre-processing sum of positive ampere hour and negative ampere hour. It will not affect the device setting values and will not trigger a new measurement operation. It only requires to get the latest available value that has been measured.

Command Syntax

FETCh[:SCALar]:ENERgy:CHARge[:SUM]?

Returned value

<NR2>

MEASure[:SCALar]:ENERgy:CHARge[:SUM]?

This command will trigger the measurement operation to get a real-time sum of positive ampere hour and negative ampere hour.

Command Syntax

MEASure[:SCALar]:ENERgy:CHARge[:SUM]?

Returned value

<NR2>

FETCh[:SCALar]:ENERgy:CHARge:POSitive?

This command is used to read the latest pre-processing positive ampere hour. It will not affect the device setting values and will not trigger a new measurement operation. It only requires to get the latest available value that has been measured.

Command Syntax

FETCh[:SCALar]:ENERgy:CHARge:POSitive?

Returned value

<NR2>

MEASure[:SCALar]:ENERgy:CHARge:POSitive?

This command will trigger the measurement operation to get a real-time positive ampere hour.

Command Syntax

MEASure[:SCALar]:ENERgy:CHARge:POSitive?

Returned value

<NR2>

FETCh[:SCALar]:ENERgy:CHARge:NEGative?

This command is used to read the latest pre-processing negative ampere hour. It will not affect the device setting values and will not trigger a new measurement operation. It only requires to get the latest available value that has been measured.

Command Syntax

FETCh[:SCALar]:ENERgy:CHARge:NEGative?

Returned value

<NR2>

MEASure[:SCALar]:ENERgy:CHARge:NEGative?

This command will trigger the measurement operation to get a real-time negative ampere hour.

Command Syntax

MEASure[:SCALar]:ENERgy:CHARge:NEGative?

Returned value

<NR2>

FETCh[:SCALar]:ENERgy:TIME?

This command is used to read the latest pre-processing integration time. It will not affect the device setting values and will not trigger a new measurement

operation. It only requires to get the latest available value that has been measured.

Command Syntax

FETCh[:SCALar]:ENERgy:TIME?

Returned value

<NR2>

MEASure[:SCALar]:ENERgy:TIME?

This command will trigger the measurement operation to get a real-time intergration time.

Command Syntax

MEASure[:SCALar]:ENERgy:TIME?

Returned value

<NR2>

FETCh[:SCALar]:ENERgy[:ACTive]:AVERage?

This command is used to read the latest pre-processing active power integral value. It will not affect the device setting values and will not trigger a new measurement operation. It only requires to get the latest available value that has been measured.

Command Syntax

FETCh[:SCALar]:ENERgy[:ACTive]:AVERage?

Returned value

<NR2>

MEASure[:SCALar]:ENERgy[:ACTive]:AVERage?

This command will trigger the measurement operation to get a real-time active power integral value.

Command Syntax

MEASure[:SCALar]:ENERgy[:ACTive]:AVERage?

Returned value

<NR2>

Chapter8 Harmonic Measurement Commands



NOTE

IT9121E do not support harmonic measurement function.

FETCh[:SCALar]:HARMonics:CURRent:AMPLitude?

This command is used to read the latest pre-processing current harmonics. It will not affect the device setting values and will not trigger a new measurement operation. It only requires to get the latest available value that has been measured.

Command Syntax

FETCh[:SCALar]:HARMonics:CURRent:AMPLitude?

Parameters

FUNDamental|TOTal|ALL / NR1

Returned value

<NRf>

MEASure[:SCALar]:HARMonics:CURRent:AMPLitude?

This command will trigger the measurement operation to get a real-time current harmonics.

Command Syntax

MEASure[:SCALar]:HARMonics:CURRent:AMPLitude?

Parameters

FUNDamental|TOTal|ALL / NR1

Returned value

<NRf>

FETCh[:SCALar]:HARMonics:CURRent:FUNDamental?

This command is used to read the latest pre-processing current fundamental harmonic. It will not affect the device setting values and will not trigger a new measurement operation. It only requires to get the latest available value that has been measured.

Command Syntax

FETCh[:SCALar]:HARMonics:CURRent:FUNDamental?

Returned value

<NR2>

MEASure[:SCALar]:HARMonics:CURRent:FUNDamental?

This command will trigger the measurement operation to get a real-time current fundamental harmonic.

Command Syntax

MEASure[:SCALar]:HARMonics:CURRent:FUNDamental?

Returned value

<NR2>

FETCh[:SCALar]:HARMonics:CURRent:THARmonic?

This command is used to read the latest pre-processing total harmonic current .It will not affect the device setting values and will not trigger a new measurement operation.It only requires to get the latest available value that has been measured.

Command Syntax

FETCh[:SCALar]:HARMonics:CURRent:THARmonic?

Returned value

<NR2>

MEASure[:SCALar]:HARMonics:CURRent:THARmonic?

This command will trigger the measurement operation to get a real-time total harmonic current.

Command Syntax

MEASure[:SCALar]:HARMonics:CURRent:THARmonic?

Returned value

<NR2>

FETCh[:SCALar]:HARMonics:CURRent:THDistort?

This command is used to read the latest pre-processing total harmonic distortion factor of current.It will not affect the device setting values and will not trigger a new measurement operation.It only requires to get the latest available value that has been measured.

Command Syntax

FETCh[:SCALar]:HARMonics:CURRent:THDistort?

Returned value

<NR2>

MEASure[:SCALar]:HARMonics:CURRent:THDistort?

This command will trigger the measurement operation to get a real-time total harmonic distortion factor of current.

Command Syntax

MEASure[:SCALar]:HARMonics:CURRent:THDistort?

Returned value

<NR2>

FETCh[:SCALar]:HARMonics:VOLTage:AMPLitude?

This command is used to read the latest pre-processing total harmonic distortion factor of voltage. It will not affect the device setting values and will not trigger a new measurement operation. It only requires to get the latest available value that has been measured.

Command Syntax

FETCh[:SCALar]:HARMonics:VOLTage:AMPLitude?

Parameters

FUNDamental|TOTal|ALL / NR1

Returned value

<NRf>

MEASure[:SCALar]:HARMonics:VOLTage:AMPLitude?

This command will trigger the measurement operation to get a real-time total harmonic distortion factor of voltage.

Command Syntax

MEASure[:SCALar]:HARMonics:VOLTage:AMPLitude?

Parameters

FUNDamental|TOTal|ALL / NR1

Returned value

<NRf>

FETCh[:SCALar]:HARMonics:VOLTage:FUNDamental?

This command is used to read the latest pre-processing voltage fundamental harmonic. It will not affect the device setting values and will not trigger a new measurement operation. It only requires to get the latest available value that has been measured.

Command Syntax

FETCh[:SCALar]:HARMonics:VOLTage:FUNDamental?

Returned value

<NR2>

MEASure[:SCALar]:HARMonics:VOLTage:FUNDament al?

This command will trigger the measurement operation to get a real-time voltage fundamental harmonic.

Command Syntax

MEASure[:SCALar]:HARMonics:VOLTage:FUNDamental?

Returned value

<NR2>

FETCh[:SCALar]:HARMonics:VOLTage:THARmonic?

This command is used to read the latest pre-processing total harmonic voltage. It will not affect the device setting values and will not trigger a new measurement operation. It only requires to get the latest available value that has been measured.

Command Syntax

FETCh[:SCALar]:HARMonics:VOLTage:THARmonic?

Returned value

<NR2>

MEASure[:SCALar]:HARMonics:VOLTage:THARmonic ?

This command will trigger the measurement operation to get a real-time total harmonic voltage.

Command Syntax

MEASure[:SCALar]:HARMonics:VOLTage:THARmonic?

Returned value

<NR2>

FETCh[:SCALar]:HARMonics:VOLTage:THDistort?

This command is used to read the latest pre-processing total harmonic distortion factor of voltage. It will not affect the device setting values and will not trigger a new measurement operation. It only requires to get the latest available value that has been measured.

Command Syntax`FETCh[:SCALar]:HARMonics:VOLTag:e:THDistort?`**Returned value**`<NR2>`**MEASure[:SCALar]:HARMonics:VOLTag:e:THDistort?**

This command will trigger the measurement operation to get a real-time total harmonic distortion factor of voltage.

Command Syntax`MEASure[:SCALar]:HARMonics:VOLTag:e:THDistort?`**Returned value**`<NR2>`**FETCh[:SCALar]:HARMonics:POWer[:ACTive]:AMPLitude?**

This command is used to read the latest pre-processing power harmonics. It will not affect the device setting values and will not trigger a new measurement operation. It only requires to get the latest available value that has been measured.

Command Syntax`FETCh[:SCALar]:HARMonics:POWer[:ACTive]:AMPLitude?`**Parameters**`FUNDamental|TOTal|ALL / NR1`**Returned value**`<NRf>`**MEASure[:SCALar]:HARMonics:POWer[:ACTive]:AMP Litude?**

This command will trigger the measurement operation to get a real-time power harmonics.

Command Syntax`MEASure[:SCALar]:HARMonics:POWer[:ACTive]:AMPLitude?`**Parameters**`FUNDamental|TOTal|ALL / NR1`**Returned value**`<NRf>`

FETCh[:SCALar]:HARMonics:POWer[:ACTive]:FUNDa mental?

This command is used to read the latest pre-processing power fundamental harmonic. It will not affect the device setting values and will not trigger a new measurement operation. It only requires to get the latest available value that has been measured.

Command Syntax

```
FETCh[:SCALar]:HARMonics:POWer[:ACTive]:FUNDa  
mental?
```

Returned value

```
<NR2>
```

MEASure[:SCALar]:HARMonics:POWer[:ACTive]:FUN Damental?

This command will trigger the measurement operation to get a real-time power fundamental harmonic.

Command Syntax

```
MEAS[:SCALar]:HARMonics:POWer[:ACTive]:FUNDa  
mental?
```

Returned value

```
<NR2>
```

FETCh[:SCALar]:HARMonics:POWer[:ACTive]:THARm onic?

This command is used to read the latest pre-processing total harmonic power. It will not affect the device setting values and will not trigger a new measurement operation. It only requires to get the latest available value that has been measured.

Command Syntax

```
FETCh[:SCALar]:HARMonics:POWer[:ACTive]:THARm  
onic?
```

Returned value

```
<NR2>
```

MEASure[:SCALar]:HARMonics:POWer[:ACTive]:THA Rmonic?

This command will trigger the measurement operation to get a real-time total harmonic power.

Command Syntax

```
MEASure[:SCALar]:HARMonics:POWer[:ACTive]:THARm  
onic?
```

Returned value

<NR2>

FETCh[:SCALar]:HARMonics:POWer[:ACTive]:THDisto rt?

This command is used to read the latest pre-processing total harmonic distortion factor of power. It will not affect the device setting values and will not trigger a new measurement operation. It only requires to get the latest available value that has been measured.

Command Syntax

FETCh[:SCALar]:HARMonics:POWer[:ACTive]:THDistort?

Returned value

<NR2>

MEASure[:SCALar]:HARMonics:POWer[:ACTive]:THDi stort?

This command will trigger the measurement operation to get a real-time total harmonic distortion factor of power.

Command Syntax

MEASure[:SCALar]:HARMonics:POWer[:ACTive]:THDistort?

Returned value

<NR2>

FETCh[:SCALar]:HARMonics:POWer:APParent?

This command is used to read the latest pre-processing apparent power harmonics. It will not affect the device setting values and will not trigger a new measurement operation. It only requires to get the latest available value that has been measured.

Command Syntax

FETCh[:SCALar]:HARMonics:POWer:APParent?

Parameters

NR1(0-50)

Returned value

<NRf>

MEASure[:SCALar]:HARMonics:POWer:APParent?

This command will trigger the measurement operation to get a real-time apparent power harmonics.

Command Syntax

MEASure[:SCALar]:HARMonics:POWer:APParent?

Parameters

NR1(0-50)

Returned value

<NRf>

FETCh[:SCALar]:HARMonics:POWer:REACtive?

This command is used to read the latest pre-processing reactive power harmonics. It will not affect the device setting values and will not trigger a new measurement operation. It only requires to get the latest available value that has been measured.

Command Syntax

FETCh[:SCALar]:HARMonics:POWer:REACtive?

Parameters

NR1(0-50)

Returned value

<NRf>

MEASure[:SCALar]:HARMonics:POWer:REACtive?

This command will trigger the measurement operation to get a real-time reactive power harmonics.

Command Syntax

MEASure[:SCALar]:HARMonics:POWer:REACtive?

Parameters

NR1(0-50)

Returned value

<NRf>

FETCh[:SCALar]:HARMonics:POWer:PFACTOR?

This command is used to read the latest pre-processing power factor harmonics. It will not affect the device setting values and will not trigger a new measurement operation. It only requires to get the latest available value that has been measured.

Command Syntax

FETCh[:SCALar]:HARMonics:POWer:PFACTOR?

Parameters

NR1(0-50)

Returned value

<NRf>

MEASure[:SCALar]:HARMonics:POWer:PFACTOR?

This command will trigger the measurement operation to get a real-time power factor harmonics.

Command Syntax

MEASure[:SCALar]:HARMonics:POWer:PFACTOR?

Parameters

NR1(0-50)

Returned value

<NRf>

FETCh[:SCALar]:HARMonics:POWer:PHASe:UU?

This command is used to read the latest pre-processing phase difference of K-order harmonic voltage(U_k) and fundamental wave(U₁).It will not affect the device setting values and will not trigger a new measurement operation.It only requires to get the latest available value that has been measured.

Command Syntax

FETCh[:SCALar]:HARMonics:POWer:PHASe:UU?

Parameters

NR1(0-50)

Returned value

<NRf>

MEASure[:SCALar]:HARMonics:POWer:PHASe:UU?

This command will trigger the measurement operation to get a real-time phase difference of K-order harmonic voltage(U_k) and fundamental wave(U₁).

Command Syntax

MEASure[:SCALar]:HARMonics:POWer:PHASe:UU?

Parameters

NR1(0-50)

Returned value

<NRf>

FETCh[:SCALar]:HARMonics:POWer:PHASe:UI?

This command is used to read the latest pre-processing phase difference of K-order harmonic voltage(Uk) and harmonic current. It will not affect the device setting values and will not trigger a new measurement operation. It only requires to get the latest available value that has been measured.

Command Syntax

FETCh[:SCALar]:HARMonics:POWer:PHASe:UI?

Parameters

NR1(0-50)

Returned value

<NRf>

MEASure[:SCALar]:HARMonics:POWer:PHASe:UI?

This command will trigger the measurement operation to get a real-time phase difference of K-order harmonic voltage(Uk) and harmonic current.

Command Syntax

MEASure[:SCALar]:HARMonics:POWer:PHASe:UI?

Parameters

NR1(0-50)

Returned value

<NRf>

FETCh[:SCALar]:HARMonics:POWer:PHASe:II?

This command is used to read the latest pre-processing phase difference of K-order harmonic current(Ik) and fundamental wave(I1). It will not affect the device setting values and will not trigger a new measurement operation. It only requires to get the latest available value that has been measured.

Command Syntax

FETCh[:SCALar]:HARMonics:POWer:PHASe:II?

Parameters

NR1(0-50)

Returned value

<NRf>

MEASure[:SCALar]:HARMonics:POWer:PHASe:II?

This command will trigger the measurement operation to get a real-time phase difference of K-order harmonic current(Ik) and fundamental wave(I1).

Command Syntax

MEASure[:SCALar]:HARMonics:POWer:PHASe:II?

Parameters

NR1(0-50)

Returned value

<NRf>

Chapter9 Sense Relevant Commands

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

This command is used to enable or disable the average function.

Command Syntax

AVERAge <bool>

Parameters

0|1|OFF|ON

Query Syntax

[[:SENSe]AVERAge:STATe?

[[:SENSe]AVERAge:TCONtrol <name>

This command is used to set the linear averaging type.

Command Syntax

AVERAge:TCONtrol <name>

Parameters

REPeat | MOVing

Query Syntax

AVERAge:TCONtrol?

[[:SENSe]AVERAge:TYPE <boolean>

This command is used to set the mode of the averaging function.

Command Syntax

AVERAge:TYPE <boolean>

Parameters

EXP|LINE

Query Syntax

AVERAge:TYPE?

[[:SENSe]AVERAge:COUNt <n>

This command is used to set the times of the averaging function.

Command Syntax

AVERAge:COUNt <n>

Parameters

1~64

Query Syntax

AVERage:COUNT?

[[:SENSE]CURRENT:RANGE:AUTO <bool>

This command is used to enable or disable the current auto range function.

Command Syntax

CURRENT:RANGE:AUTO <bool>

Parameters

0|1|OFF|ON

Query Syntax

CURRENT:RANGE:AUTO?

[[:SENSE]VOLTage:RANGE:AUTO <bool>

This command is used to enable or disable the voltage auto range function.

Command Syntax

VOLTage:RANGE:AUTO <bool>

Parameters

0|1|OFF|ON

Query Syntax

VOLTage:RANGE:AUTO?

[[:SENSE]CURRENT:RANGE <NRf>

This command is used to set the current range.

Command Syntax

CURRENT:RANGE <NRf>

Parameters

Model	Current Range
IT9121	CF=3: 5mA, 10mA, 20mA, 50mA, 100mA, 200mA, 500mA, 1A, 2A, 5A, 10A, 20A, Auto CF=6: 2.5mA, 5mA, 10mA, 25mA, 50mA, 100mA, 250mA, 0.5A, 1A, 2.5A, 5A, 10A, Auto
IT9121E	CF=3: 5mA, 10mA, 20mA, 50mA, 100mA, 200mA, 500mA, 1A, 2A, 5A, 10A, 20A, Auto CF=6: 2.5mA, 5mA, 10mA, 25mA, 50mA, 100mA, 250mA, 0.5A, 1A, 2.5A, 5A, 10A, Auto
IT9121H	CF=3: 5mA, 10mA, 20mA, 50mA, 100mA, 200mA, 500mA, 1A,

	2A, 5A, 10A, 20A, Auto CF=6: 2.5mA, 5mA, 10mA, 25mA, 50mA, 100mA, 250mA, 0.5A, 1A, 2.5A, 5A, 10A, Auto
IT9121C	CF=3: 1A, 2A, 5A, 10A, 20A, 50A, Auto CF=6: 0.5A, 1A, 2.5A, 5A, 10A, 25A, Auto

Query Syntax

CURRent:RANGe?

[[:SENSE]VOLTage:RANGe <NRf>

This command is used to set the voltage range.

Command Syntax

VOLTage:RANGe <NRf>

Parameters

Model	Voltage Range
IT9121	CF=3: 15V, 30V, 60V, 150V, 300V, 600V, Auto CF=6: 7.5V, 15V, 30V, 75V, 150V, 300V, Auto
IT9121E	CF=3: 15V, 30V, 60V, 150V, 300V, 600V, Auto CF=6: 7.5V, 15V, 30V, 75V, 150V, 300V, Auto
IT9121H	CF=3: 15V, 30V, 60V, 150V, 300V, 600V, 1000V(CF=1.5), Auto CF=6: 7.5V, 15V, 30V, 75V, 150V, 300V, 500V(CF=3), Auto
IT9121C	CF=3: 15V, 30V, 60V, 150V, 300V, 600V, Auto CF=6: 7.5V, 15V, 30V, 75V, 150V, 300V, Auto

Query Syntax

VOLTage:RANGe?

**[[:SENSE]CURRent:EXS1[:STATE] **

This command is used to enable or disable the external current sensor 1.

Command Syntax

CURRent:EXS1[:STATE]

Parameters

OFF|ON

Query Syntax

CURRent:EXS1:STATE?

**[[:SENSE]CURRent:EXS2[:STATE] **

This command is used to enable or disable the external current sensor 2.

Command Syntax

CURRent:EXS2[:STATE]

Parameters

OFF|ON

Query Syntax

CURRent:EXS2:STATe?

[[:SENSEe]CURRent:SRATio:EXS1 <NRf>

The command is used to set the conversion ratio of the external current sensor 1.

Command Syntax

CURRent:SRATio:EXS1 <NRf>

Parameters

0.001~9999.999

Query Syntax

CURRent:SRATio:EXS1?

[[:SENSEe]CURRent:SRATio:EXS2 <NRf>

The command is used to set the conversion ratio of the external current sensor 2.

Command Syntax

CURRent:SRATio:EXS2 <NRf>

Parameters

0.001~9999.999

Query Syntax

CURRent:SRATio:EXS2?

[[:SENSEe:]CURRent:EXS:RANGe

This command is used to set the current range of external sensor. Default unit is V.

Command Syntax

CURRent:EXS:RANGe <NRf>

Parameters

Model	Current range of external sensor
IT9121	CF=3: 2.5V, 5V, 10V, 50mV, 100mV, 200mV, 500mV, 1V, 2V CF=6: 1.25V, 2.5V, 5V, 25mV, 50mV, 100mV, 250mV, 0.5V, 1V
IT9121E	CF=3: 2.5V, 5V, 10V, 50mV, 100mV, 200mV, 500mV, 1V, 2V CF=6: 1.25V, 2.5V, 5V, 25mV, 50mV, 100mV, 250mV, 0.5V, 1V
IT9121H	CF=3: 2.5V, 5V, 10V, 50mV, 100mV, 200mV, 500mV, 1V, 2V

	CF=6: 1.25V, 2.5V, 5V, 25mV, 50mV, 100mV, 250mV, 0.5V, 1V
IT9121C	CF=3: 5V, 10V, 100mV, 250mV, 500mV, 1V, 2.5V CF=6: 2.5V, 5V, 50mV, 125mV, 250mV, 0.5V, 1.25V

Query Syntax

CURRent:EXS:RANGe?

Chapter10 Trigger Commands

ABORt

This command set related sensors in idle condition. When measuring process is terminated, it will start a new self trigger system of measurement without affected.

Command Syntax

ABORt

Parameters

None

Query Syntax

None

INITiate:CONTInuous <state>

This command is used to enable or disable the state of continuous measurement period. In off mode, that means to enable a single measurement.

Command Syntax

INITiate:CONTInuous <state>

Parameters

ON/OFF

Query Syntax

INITiate:CONTInuous?

INITiate[:IMMediate]

This command will create a single measurement operation and relevant sensors are in request status.

Command Syntax

INITiate:IMMediate

Parameters

None

Query Syntax

None

TRIGger:IMMediate

This command is available for all trigger mode. It will generate a trigger signal.

Command Syntax

TRIGger:IMMEDIATE

Parameters

None

Relevant command

*TRG TRIG:SOUR

TRIGger:SOURce <state>

This command is used to select trigger source. Trigger source modes are as below. When execute *RST command, trigger source should be set in MANUAL mode.

Parameters	Function description
IMMEDIATE	
BUS	Sending TRIG:IMM or *TRG command will create a trigger signal.
EXTERNAL	External clock trigger mode
VOLTage	Set input voltage source as a trigger signal
CURRENT	Set input current source as a trigger signal

Command Syntax

TRIGger:SOURce <state>

Parameters

IMMEDIATE|BUS|EXTERNAL|VOLTage|CURRENT

Query Syntax

TRIGger:SOURce?

TRIGger:SLOPe <state>

This command is used to set the trigger slope, parameters could be rising edge/descending dege/any edge.

Command Syntax

TRIGger:SLOPe <state>

Parameters

POSitive|NEGative|ANY

Query Syntax

TRIGger:SLOPe?

TRIGger:VOLTage:LEVel <level>

This command is used to set the voltage trigger level when Voltage selected as

trigger source mode.

Command Syntax

TRIGger:VOLTage:LEVel <level>

Parameters

<level>

Query Syntax

TRIGger:VOLTage:LEVel?

TRIGger:CURRent:LEVel <level>

This command is used to set the current trigger level when Current selected as trigger source mode.

Command Syntax

TRIGger:CURRent:LEVel <level>

Parameters

level

Query Syntax

TRIGger:CURRent:LEVel?

Chapter11 Oscilloscope Commands

WAVE:TRIG:SOUR < VOLTage|CURRent|EXTernal >

This command can set or read the device taking which kind waveform as trigger source.

Command Syntax

WAVE:TRIG:SOUR < VOLTage|CURRent|EXTernal >

Parameters

VOLTage|CURRent|EXTernal

Query Syntax

WAVE:TRIG:SOUR?

WAVE:TRIG:SLOP < POSitive|NEGative|ANY >

This command is used to set trigger slope when takes waveform as trigger source.

Command Syntax

WAVE:TRIG:SLOP < POSitive|NEGative|ANY >

Parameters

POSitive|NEGative|ANY

Query Syntax

WAVE:TRIG:SLOP?

WAVE:TRIG:MODE < AUTO|NORMal >

This command is used to set the trigger mode when take waveform as trigger source.

Command Syntax

WAVE:TRIG:MODE < AUTO|NORMal >

Parameters

AUTO|NORMal

Query Syntax

WAVE:TRIG:MODE?

WAVE:TRIG:DELay:TIME <n>

This time is used to set the trigger delay time when take a waveform as trigger source.(Note:the delay time setting should be less than the time width of the whole screen.

Command Syntax

WAVE:TRIG:DElay:TIME <n>

Parameters

<n>

Unit

S

Query Syntax

WAVE:TRIG:DElay:TIME?

WAVE:TRIG:DIVTime (UNIT:S) <n>

This command is used to set the time/grid value of the waveform display.
(Available setting:0.0005,0.001,0.002,0.005,.0.01,0.02.,0.05,0.1,0.2,0.5)

Command Syntax

WAVE:TRIG:DIVTime (UNIT:S) <n>

Parameters

<n>

Unit

S

Query Syntax

WAVE:TRIG:DIVTime?

WAVE:RUN

This command is used to start waveform capture.

Command Syntax

WAVE:RUN

Parameters

None

Query Syntax

None

WAVE:STOP

This command can stop the waveform capture.

Command Syntax

WAVE:STOP

Parameters

None

Query Syntax

None

WAVE:SINGLE

This command is used to trigger a single waveform capture.

Command Syntax

WAVE:SINGLE

Parameters

None

Query Syntax

None

WAVE:VOLTage:DATA[:NORMAlization]?

This command is used to obtain the voltage datas after normalization.

Command Syntax

WAVE:VOLTage:DATA?

Parameters

<n>

Query Syntax

WAVE:VOLTage:DATA?

WAVE:CURREnt:DATA[:NORMAlization]?

This command is used to obtain the current datas after normalization.

Command Syntax

WAVE:CURREnt:DATA[:NORMAlization]?

Parameters

<n>

Query Syntax

WAVE:CURREnt:DATA[:NORMAlization]?

WAVE:TRIGger[:STATe]?

This command is used to query the trigger status.

Command Syntax

WAVE:TRIGger[:STATe]?

Returned value

Auto| Auto?|Trig|Trig?|Stop

Query Syntax

WAVE:TRIGger[:STATe]?

Chapter12 INPut Commands

[[:INPut]:HARMonics:ORDer <NRI>

This command is used to set the harmonic analyse orders.

Command Syntax

HARMonics:ORDer <NRI>

Parameters

2-50

Query Syntax

HARMonics:ORDer?

[[:INPut]:HARMonic:PLLSource <OFF|U|I>

This command set PLL (Phase Locked Loop)source which is used for determining the fundamental wave cycle as the reference for analysis of harmonic orders.

Command Syntax

HARMonic:PLLSource <OFF|U|I>

Parameters

OFF|U|I

Query Syntax

HARMonic:PLLSource?

[[:INPut]:HARMonics:THD <THDR|THDF>

This command is used to set the calculation formula of THD(Total Harmonic Distrotion).

Command Syntax

HARMonics:THD <THDR|THDF>

Parameters

THDR|THDF

Query Syntax

HARMonics:THD?

[[:INPut]:HARMonic:SEQuence <ALL|ODD|EVEN>

This command is used to select the harmonic sequence of ALL/ODD/EVEN.

Command Syntax

HARMonic:SEQuence <ALL|ODD|EVEN >

Parameters

ALL|ODD|EVEN

Query Syntax

HARMonic:SEQuence?

[:INPut]:SSource <OFF|U|I>

This command is used to set the synchronous source type.

Command Syntax

SSource <OFF|U|I>

Parameters

OFF|U|I

Query Syntax

SSource?

[:INPut]:CFACtor <NRI>

This command is used to set the crest factor.

Command Syntax

CFACtor <NRI>

Parameters

3|6

Query Syntax

CFACtor?

[:INPut]:FILTer:FREQuency <bool>

This command is used to set the state of frequency filter.

Command Syntax

FILTer:FREQuency <bool>

Parameters

0|1|OFF|ON

Query Syntax

FILTer:FREQuency?

[[:INPut]:]FILTEr:LINE <bool>

This command is used to set the state of line filter.

Command Syntax

FILTEr:LINE <bool>

Parameters

0|1|OFF|ON

Query Syntax

FILTEr:LINE?

[[:INPut]:]RATE <NRI>

This command is used to set the data updating rate.

Command Syntax

RATE <NRI>

Parameters

0.1s/0.25s/0.5s/1s/2s/5s

Query Syntax

RATE?

[INPut:]:INTEgral:ACAL <bool>

This command is used to enable or disable the auto calculation function of integral.

Command Syntax

INTEgral:ACAL <bool>

Parameters

0|1|OFF|ON

Query Syntax

INTEgral:ACAL?

[INPut]:INTEgral:QMODE {RMS|MN|DC|RMN|AC}

This command is used to set the current integral mode.

Command Syntax

INTEgrate:QMODE {RMS|MN|DC|RMN|AC}

Parameters

RMS|MN|DC|RMN|AC

Query Syntax

INTEgrate:QMODE?

[INPut]:INTEgral:WPTYpe {CHARge|SOLD|DISCharge|BOUGht }

This command is used to set integral mode of watt hour.

Command Syntax

INTEgrate:WPTYpe {CHARge|SOLD|DISCharge|BOUGht }

Parameters

CHARge|SOLD|DISCharge|BOUGht

Query Syntax

INTEgrate:WPTYpe?

[INPut:]INRush[:STATe] <OFF|ON>

This command is used to enable or disable the inrush current measurement function.

Command Syntax

[INPut:]INRush[:STATe] <OFF|ON>

Parameters

OFF|ON

Query Syntax

INRush?

[INPut:]INRush:TRIGger:CURRent[:LEVel] <n>

This command is used to set the trigger level for inrush current measurement.

Command Syntax

[INPut:]INRush:TRIGger:CURRent[:LEVel] <n>

Parameters

<n>

Unit

A

Query Syntax

INRush:TRIGger:CURRent?

[INPut:]INRush:DELay:TIME <n>

This command is used to set the delay time of inrush current measurement.(Max time is 0.0002s).

Command Syntax

[INPut:]INRush:DELay:TIME <n>

Parameters

<n>

Unit

S

Query Syntax

INRush:DELay:TIME?

[INPut:]INRush:MEASure:TIME <n>

This command is used to set the inrush current measuring time.(max time is 300s).

Command Syntax

[INPut:]INRush:MEASure:TIME <n>

Parameters

<n>

Unit

S

Query Syntax

INRush:MEASure:TIME?

Chapter13 CALCulate Commands

[CALCulate]:INTEgral[:STATe]

This command is used to state of integral function.

Command Syntax

CALCulate:INTEgral[:STATe]

Parameter

ON|OFF

Query Syntax

CALCulate:INTEgral?

[CALCulate]:INTEgral:CLEAr[:IMMediate]

This command is used to clear all integral value.

Command Syntax

CALCulate:INTEgral:CLEAr

Parameters

None

Query Syntax

None

[CALCulate]:INTEgral:CLEAr:AUTO <ON|OFF>

This command is used to enable or disable the automatic zero clearing function.

Command Syntax

INTEgral:CLEAr:AUTO <ON|OFF>

Parameters

ON|OFF

Query Syntax

INTEgral:CLEAr:AUTO?

[CALCulate]:INTEgral:STARt:SOURce < TIME|MAN>

This command is used to define the mode how to start the integral function.

Time:start to integral on the set date

MAN:press the “START” soft key in the integral measurement interface to trigger the manual start of the integral function.

Command Syntax

INTEgral:START:SOURce < TIME|MAN>

Parameters

TIME|MAN

Query Syntax

INTEgral:START:SOURce?

[[:CALCulate]:INTEgral:STARt[:IMMediate]

When integral start mode is set to CMD,sending this command will trigger integral operation.

Command Syntax

INTEgral:STARt

Parameters

None

Query Syntax

None

[[:CALCulate]:INTEgral:STARt:DATE<yy,MM,dd>

[[:CALCulate]:INTEgral:STARt:TIME <hh,mm,ss>

This command is used to set the start date,when time reached,device start integral operation.

Command Syntax

INTEgral:STARt:DATE <yy,MM,dd>
INTEgral:STARt:TIME <hh,mm,ss>

Parameters

<yy,MM,dd> <hh,mm,ss>

Query Syntax

INTEgral:STARt:DATE ?
INTEgral:STARt:TIME ?

[[:CALCulate]:INTEgral:STOP:SOURce<TIME|MAN|TINterval>

This command is used to set the integral stop mode.

Command Syntax

INTEgral:STOP:SOURce < TIME|MAN|TINterval>

Parameters

TIME|MAN|TINterval

Query Syntax

INTEgral:STOP:SOURce?

[[:CALCulate]:INTEgral:STOP[:IMMediate]

When integral stop mode is set to CMD, sending this command will stop the integral operation. When stopped, integral value will not be cleared to zero.

Command Syntax

INTEgral:STOP

Parameters

None

Query Syntax

None

[[:CALCulate]:INTEgral:STOP:DATE:<yy,MM,dd>

[[:CALCulate]:INTEgral:STOP:TIME: <hh,mm,ss>

This command is used to set the stop date, when time reached, device stops integral operation.

Command Syntax

INTEgral:STOP:DATE:<yy,MM,dd>
INTEgral:STOP:TIME: <hh,mm,ss>

Parameters

<yy,MM,dd> <hh,mm,ss>

Query Syntax

INTEgral:STOP:DATE?
INTEgral:STOP:TIME?

[[:CALCulate]:INTEgral:STOP:TINterval hhhh,mm,ss >

<

When integral stop mode is set to TINterval, user can set the integral measuring period via this command. When time length reached, integral operation will be terminated.

Command Syntax

INTEgral:STOP:TINterval < hhhh,mm,ss >

Parameters

hhhh,mm,ss

Query Syntax

INTEgral:STOP:TINTerval?

INTEgral:CONDition?

This command is used to obtain the integral running state.

Command Syntax

INTEgral:CONDition?

Returned value

Ready|Start|Stop|Time up|Error

Query Syntax

INTEgral:CONDition?

[[:CALCulate]][:METer]:MAXHold[:STATE] <bool>

This command is used to set the hold mode of maximum value.

Command Syntax

CALCulate:METer:MAXHold <bool>

Parameters

0|1|OFF|ON

Query Syntax

CALCulate:METer:MAXHold?

[[:CALCulate]][:METer]:CLEar[:IMMediate]

This command is used to clear measuring value to zero.

Command Syntax

CALCulate:METer:CLEar:IMMediate

Parameters

None

Query Syntax

None

[[:CALCulate]:HARMonic[:STATE] <bool>

This command is used to set the harmonic state.

Command Syntax

CALCulate:HARMonic <bool>

Parameters

0|1|OFF|ON

Query Syntax

CALCulate:HARMonic?

[:CALCulate]:SCOPE[:STATE] <bool>

This command is used to enable or disable the oscilloscope function.

Command Syntax

CALCulate:SCOPE <bool>

Parameters

0|1|OFF|ON

Query Syntax

CALCulate:SCOPE?

[CALCulate:]HOLD[:STATE] <bool>

This command is used to enable or disable the Hold mode.

Command Syntax

[CALCulate:]HOLD[:STATE]

Parameters

0|1|OFF|ON

Query Syntax

HOLD?

Chapter14 System Commands

SYSTem:BEEPer:IMMediate

This command is used to test the beeper. The power meter will beep for once after executing this command.

Command Syntax

SYSTem:BEEPer:IMMediate

Parameters

None

Query Syntax

None

SYSTem:BEEPer[:STATe] <bool>

This command is used to turn on or turn off the beeper.

Command Syntax

SYSTem:BEEPer <bool>

Parameters

0|1|OFF|ON

Query Syntax

SYSTem:BEEPer:STATe?

SYSTem:DATE <NRf>,<NRf>,<NRf>

This command is used to set the system date.

Command Syntax

SYSTem:DATE <NRf>,<NRf>,<NRf>

Parameters format

yyyy,mm,dd

Query Syntax

SYSTem:DATE?

SYSTem:TIME <NRf>,<NRf>,<NRf>

This command is used to set the system time.

Command Syntax

SYSTem:TIME <NRf>,<NRf>,<NRf>

Parameters format

hh,mm,ss

Query Syntax

SYSTem:TIME?

SYSTem:KEY <NRi>

This command is used to query the final one of the key pressed.

Command Syntax

SYSTem:KEY?

Parameters

None

SYSTem:ERRor?

This command is used to query the error information.

Command Syntax

SYSTem:ERRor?

Parameters

None

SYSTem:LANGuage <NRi>

This command is use to set the system language.

Command Syntax

Parameters

Query Syntax

?

SYSTem:LOCal

This command is used to switch the power meter to local operation mode.

Command Syntax

SYST:LOC

Parameters

None

Query Syntax

None

SYSTem:REMOte

This command is used to switch the power meter to remote control mode. In this mode, except Esc button (pressing this button for 5s will switch the unit to local mode), other keys are locked.

Command Syntax

SYST:REM

Parameters

None

Query Syntax

None

SYSTem:CLEAr

This command is used to clear the error information.

Command Syntax

SYSTem:CLEAr

Parameters

None

Returned value

None

SYSTem:RWLock

This command is used to set power meter to remote control mode via RS232 communication interface. And Esc button is not available. Other buttons are locked too.

Command Syntax

SYST:RWL

Parameters

None

Returned value

None

SYSTem:VERSIon?

This command is used to query the device version. Return value is a character string as like "YYYY.V". YYYY represents the year and V means the version of that year.

Command Syntax

SYST:VERS?

Parameters

None

Returned value

<NR2>

Example:1991.0

Chapter15 Calibration Commands

CALibration:ZERO

Zero calibration command.

Command Syntax

CALibration:ZERO

Parameters

None

CALibrate:SECure[:STATE]

This command is used to enable or disable calibration mode.

The calibration mode must be enabled before the load will accept any other calibration commands. The first parameter specifies the enabled or disabled state. The second parameter is the password. It is required if the calibration mode is being enabled and the existing password is not 0. If the password is not entered or is incorrect, an error is generated and the calibration mode remains disabled. The query statement returns only the state, not the password. Whenever the calibration state is changed from enabled to disabled, any new calibration constants are lost unless they have been stored with CALibrate:SAVE.

Command Syntax

CALibrate:SECure[:STATE] <bool> [,<SRD>]

Parameters

0 | 1 | OFF | ON [,<password>]

Reset value

ON

Example

CAL:SEC 0, N3301A CAL:SEC ON

Query Syntax

CALibrate:SECure[:STATE]?

Parameters

<NR1>

Relevant command

CAL:SAVE CAL:INIT

CALibrate:INITial

This command can only be used in calibration mode. It restores factory

calibration constants from nonvolatile memory.

Command Syntax

CALibrate:INITial

Parameters

None

Example

CAL:INIT

Relevant command

CAL:STAT CAL:INIT
CHAN:FACTory:SAVE

Command Syntax

CHAN:FACTory:SAVE

Parameters

None

CALibrate:SAVE

This command can only be used in calibration mode. It saves any new calibration constants (after a current or voltage calibration procedure has been completed) in nonvolatile memory.

Command Syntax

CALibrate:SAVE

Parameters

None
Example:CAL:SAVE

Relevant command

CAL:STAT CAL:INIT

CALibrate:CURRENT:POINT

This command is used to specify the current calibration points.

Command Syntax:
CALibrate:CURRENT:POINT <point>

Parameters

P1 | P2 | P3 | P4
Example:CAL:CURRE:POIN P2

Relevant command

CAL:STAT CAL:SAV

CALibrate:CURRent[:LEVel]

This command only used for calibration mode.

It enters a calibration voltage value that you obtain by reading an external meter. You must first select a calibration level (with CALibrate:VOLTagE:POINT) for the value being entered. These constants are not stored in nonvolatile memory until they are saved with CALibrate:SAVE.

Command Syntax

CALibrate:CURRent[:LEVel] <NRf>

Parameters

external reading

Unit:A (amps)

Example:CAL:CURR 3.2223

Relevant command

CAL:STAT CAL:SAV

CALibrate:CURRent:EXTernal:POINT<point>

This command is used to calibrate external current points.

Command Syntax

CALibrate:CURRent:EXTernal:POINT<point>

Parameters

<point>

CALibrate:CURRent:EXTernal:[LEVel] <NRf>

This command is used to input external calibration current.

Command Syntax

CALibrate:CURRent:EXTernal:[LEVel] <NRf>

Parameters

<NRf>

CALibrate:VOLTagE:POINT

This command can only be used in calibration mode. It is used to set the calibration points of constant voltage mode. P1, P2 is used in low voltage meter range, P3, P4 is used in high voltage meter range. It can use calibrate voltage source and voltage meter.

Command Syntax

CALibrate:VOLTagE:POINT <point>

Parameters

P1 | P2 | P3 | P4

Example:CAL:VOLT:POIN P2

Relevant command

CAL:STAT CAL:SAV

CALibrate:VOLTage[:LEVel]

This command is only used in calibration mode. It enters a calibration voltage value that you obtain by reading an external meter. You must first select a calibration level (with CALibrate:VOLTage:POINT) for the value being entered. These constants are not stored in nonvolatile memory until they are saved with CALibrate:SAVE.

Command Syntax

CALibrate:VOLTage[:LEVel] <NRf>

Parameters

external reading

UnitV (volts)
Example:CAL:VOLT 3.2223**Relevant Command**

CAL:STAT CAL:SAV

Chapter16 IEEE-488 Commands

This chapter introduced often-used IEEE-488 commands.

***CLS**

This command is used to clear following registers:

- Standard Event Register
- Query Event Register
- Status Byte Register

Command Syntax

*CLS

Parameters

None

***ESE**

This command is used to edit the value of standard event enable register. It defines the specified bits from standard event register that will cause the value of ESB bit in status byte register to be 1.

Command Syntax

*ESE <NR1>

Parameters

0~255

Power-On Value

Refer to *PSC command

Example

*ESE 128

Query Syntax

*ESE?

Returned value

<NR1>

Relevant Command

*ESR? *PSC *STB?

***ESR?**

This command is used to read the value of standard event registers. And values will be cleared to zero after executing this command. The bit definition of

standard event register and standard event enable register are the same.

Query Syntax

*ESR?

Parameters

None

Returned value

<NR1>

Relevant Command

*CLS *ESE *ESE? *OPC

*IDN?

This command is used to query related information of power meter.

Query Syntax

*IDN?

Parameters

None

Returned value

<AARD>

Example

ITECH,IT9121,KN34243232,01.00

*OPC

After all other commands executed before *OPC command, the OPC bit in standard event register will be set to 1. Sending query standard event register command will return value 1 into the output buffer.

Command Syntax

*OPC

Parameters

None

Query Syntax

*OPC?

Returned value

<NR1>

*RST

This command reset the device to factory default setup.

Command Syntax

*RST

Parameters

None

*SRE

This command sets the condition of the Status Request Enable Register. After executing this command, power meter will return back a decimal value which is a binary weighted sum of all bits from the enable register.

Command Syntax

*SRE <NRf>

Parameters

0~255

Power-on value

Refer to *PSC command

Example

*SRE 128

Query Syntax

*SRE?

Returned value

<NR1>

Relevant Command

*ESE *ESR? *PSC *STB?

*STB?

This command can query the Status Byte register. After executing this command, the bit6 of status byte register will be reset to 0.

Query Syntax

*STB?

Parameters

None

Returned value

<NR1>

Relevant Command

*CLS *ESE *ESR

***WAI**

This command instructs the power meter not to process any further commands until all pending operations are completed. Pending operations are as defined under the *OPC command. *WAI can be aborted only by sending the power meter a device clear command.

Command Syntax

*WAI

Parameters

None

Relevant Command

*OPC

Contact US

Thank you for purchasing ITECH products.If you have any doubt about this product, please contact us as follow.

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