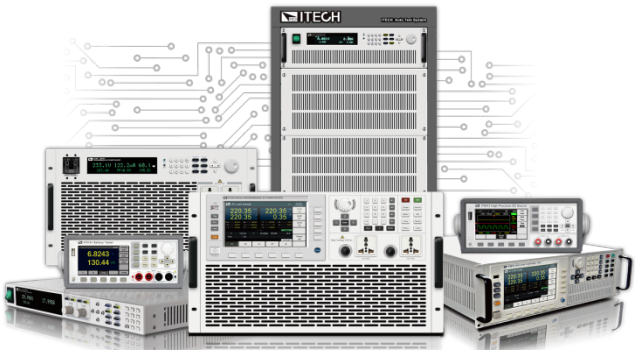
Programmable AC Power Source
IT-M7700 Programming Guide



Model: IT-M7721/IT-M7722/IT-M7721L/IT-M7722L/
IT-M7722D/IT-M7723D/IT-M7723/
IT-M7722E/IT-M7723E/IT-M7723P
Version: V1.7

Notices

Manual Part Number

IT-M7700-400000

Revision

1st Edition: December 11,

2023

Itech Electronic, Co., Ltd.

Trademarks

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

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

Warranty

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

Technology Licenses

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

Restricted Rights Legend

Restricted permissions of the U.S. government. Permissions for software and technical data which are authorized to the U.S. Government only include those for custom provision to end users. ITECH 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 IT-M7700 series power source meets all the published specifications at time of shipment from the factory.

Warranty

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

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

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

Limitation of Warranty

This Warranty will be rendered invalid 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
⏚	Chassis (earth ground) symbol.	⎎	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. Do not operate the instrument in the presence of inflammable gasses, vapors or dusts.**
- **The instrument is provided with a power cord during delivery and should be connected to a socket with a protective earth terminal, a junction box or a three-phase distribution box. Before operation, be sure that the instrument is well grounded.**
- **Please always use the provided cable to connect the instrument.**
- **Check all marks on the instrument before connecting the instrument to power supply.**
- **Ensure the voltage fluctuation of mains supply is less than 10% of the working voltage range in order to reduce risks of fire and electric shock.**
- **Do not install alternative parts on the instrument or perform any unauthorized modification.**
- **Do not use the instrument if the detachable cover is removed or loosen.**
- **To prevent the possibility of accidental injuries, be sure to use the power adapter supplied by the manufacturer only.**
- **We do not accept responsibility for any direct or indirect financial damage or loss of profit that might occur when using the instrument.**
- **This instrument is used for industrial purposes, do not apply this product to IT power supply system.**
- **Never use the instrument with a life-support system or any other equipment subject to safety requirements.**

WARNING

- **SHOCK HAZARD Ground the Instrument. This product is provided with a protective earth terminal. To minimize shock hazard, the instrument must be connected to the AC mains through a grounded power cable, with the ground wire firmly connected to an electrical ground (safety ground) at the power outlet or distribution box. Any interruption of the protective (grounding) conductor or disconnection of the protective earth terminal will cause a potential shock hazard that could result in injury or death.**
- **Before applying power, verify that all safety precautions are taken. All connections must be made with the instrument turned off, and must be performed by qualified personnel who are aware of the hazards involved. Improper actions can cause fatal injury as well as equipment damage.**
- **SHOCK HAZARD, LETHAL VOLTAGES This product can output the dangerous voltage that can cause personal injury, and the operator must always be protected from electric shock. Ensure that the output electrodes are either insulated or covered using the safety covers**

provided, so that no accidental contact with lethal voltages can occur.

- **Never touch cables or connections immediately after turning off the instrument. Verify that there is no dangerous voltage on the electrodes or sense terminals before touching them.**

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	0°C to 40°C
Operating humidity	20%-80% (non-condensation)
Storage temperature	-10°C to 70 °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

CE	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 Equiment (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 [1][2][3]

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

Chapter 1 SCPI Command Introduction

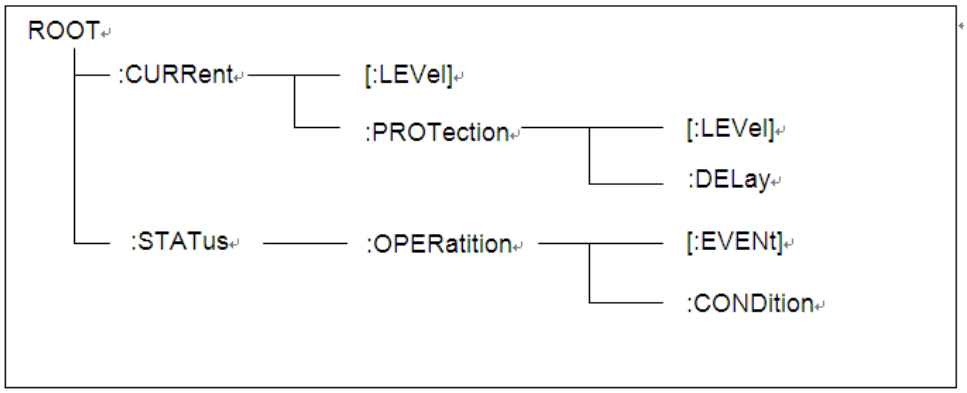
1.1 Overview

SCPI is short for Standard Commands for Programmable Instruments which defines a communication method of bus controller and instrument. It is based on ASCII and supply for testing and measuring instruments. SCPI command is based on hierarchical architecture which also known as tree system.

1.2 Command Type of SCPI

SCPI has two types of commands, common and subsystem.

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



Multiple commands in a message

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

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

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

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

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

Movement in the subsystem

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

PROTection:CLEAr;:STATus:OPERation:CONDition?

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

POWer:LEVel 200;PROTection 28; :CURRent:LEVel 3;PROTection:STATe ON

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

Including Common Commands

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

VOLTage:TRIGgered 17.5;:INITialize;*TRG

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

Case sensitivity

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

*RST = *rst

:DATA? = :data?

:SYSTem:PRESet = :system:preset

Long-form and short-form versions

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

:SYSTem:PRESet long-form

:SYST:PRES short form

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

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

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

Query

Observe the following precautions with queries:

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

1.3 Message Typeof SCPI

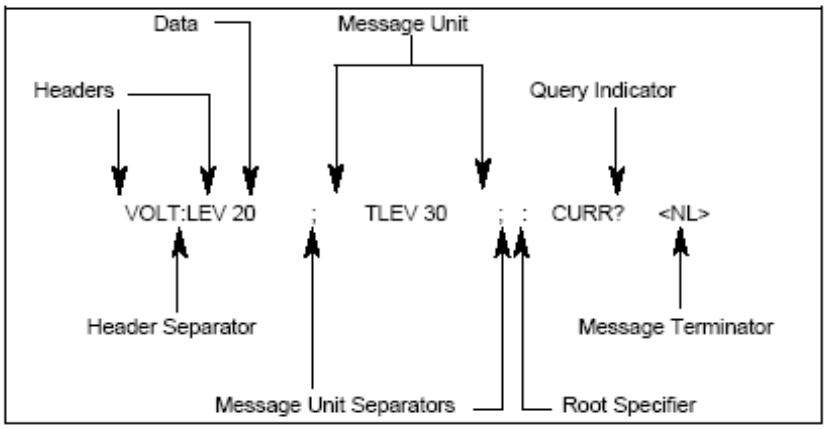
There are two types of SCPI messages, program and response.

- program message: A program message consists of one or more properly formatted SCPI commands sent from the controller to the instrument. The message, which may be sent at any time, requests the instrument to

perform some action.

- response message: A response message consists of data in a specific SCPI format sent from the instrument to the controller. The instrument sends the message only when commanded by a program message called a "query."

The next figure illustrates SCPI message structure:



The Message Unit

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

ABORt<NL>

VOLTage 20<NL>

Headers

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

Query Indicator

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

Message Unit Separator

When two or more message units are combined into a compound message,

separate the units with a semicolon (STATus:OPERation?;QUEStionable?).

Root Specifier

When it precedes the first header of a message unit, the colon becomes the root specifier.

Message Terminator

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

- newline (<NL>),decimal 10 or hexadecimal 0X0A in ASCII.
- end or identify (<END>)
- both of the above (<NL><END>).

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

Command execution rules

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

1.4 Response Data Type

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

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

Response messages

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

Sending a response message

After sending a query command, the response message is placed in the Output

Queue. When the instrument is then addressed to talk, the response message is sent from the Output Queue to the computer.

Multiple response messages

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

0; 1; 1; 0

Response message terminator (RMT)

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

0; 1; 1; 0; <RMT>

Message exchange protocol

Two rules summarize the message exchange protocol:

Rule 1.You must always tell the IT6500 Series what to send to the computer.

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

1. Send the appropriate query command(s) in a program message.

2. Address the IT6500 Series to talk.

Rule 2.The complete response message must be received by the computer before another program message can be sent to the IT6500 Series.

1.5 Command Format

Formats for command display are as follows:

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

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

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

Long-format commands may be sent to ensure better program readability.

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

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

Colon (:)

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

APPL:SIN 455E3,1.15,0.0

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

Semicolon (;)

It is used to separate several commands in the same subsystem and can also

minimize typing. For example, to send the following command string:

TRIG:SOUR EXT; COUNT 10

has the same effect as sending the following two commands:

TRIG:SOUR EXT
TRIG:COUNT 10

Question mark (?)

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

TRIG:COUN 10

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

TRIG:COUN?

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

TRIG:COUN?MIN
TRIG:COUN?MAX

Comma (,)

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

Space

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

Generic commands (*)

The IEEE-488.2 standard defines a set of common commands that perform functions such as reset, self-test, and status operations. Generic commands always start with a asterisk (*) and occupy 3 character sizes, including one or more Parameter. Key words of a command and the first parameter are separated by a space. Semicolon (;) can separate several commands as follows:

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

Command terminator

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

acceptable to place one <NL> after a <Enter>. Termination of command string always resets current SCPI command path to root level.

NOTE

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

1.6 Data Type

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

- Numerical parameter

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

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

- <NR1>: represents an integer value, such as 273;
- <NR2>: represents a real number in floating-point format, such as 0.273;
- <NR3>: represents a real number in scientific notation, such as 2.73E+2;
- <Nrf>:The extensible form includes <NR1>, <NR2> and <NR3>;
- <Nrf+>: The extensible decimal form includes <NRf> and MIN MAX DEF, MIN and MAX represent the minimum and maximum finite number. DEF is the default of the parameter.

- Discrete parameter

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

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

- Boolean parameter

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

DISPlay {OFF|0|ON|1}

- ASCII string Parameter

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

DISPlay:TEXT <quoted string>

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

DISP:TEXT "WAITING..."

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

DISP:TEXT 'WAITING...'

- <CPD>: character program data;
- <SPD>: string program data. String parameters enclosed in single or double quotes.

1.7 Communication Interface

IT-M7700 have five standard communication interfaces: RS232, USB, GPIB, LAN and CAN. The customer can choose any one according to his demands. Please refer to user manual for detailed introductions of the remote interface connections.

Chapter 2 Status Register

2.1 Status Register Definition

You can get the current status of the power supply by reading the status registers. The power supply records the different status of the instrument through the four status register group. The four status register groups are: Questionable Status Register, Operation Status Register, Standard Event Register and Status Byte Register. Status Byte Register records the information of the other status register.

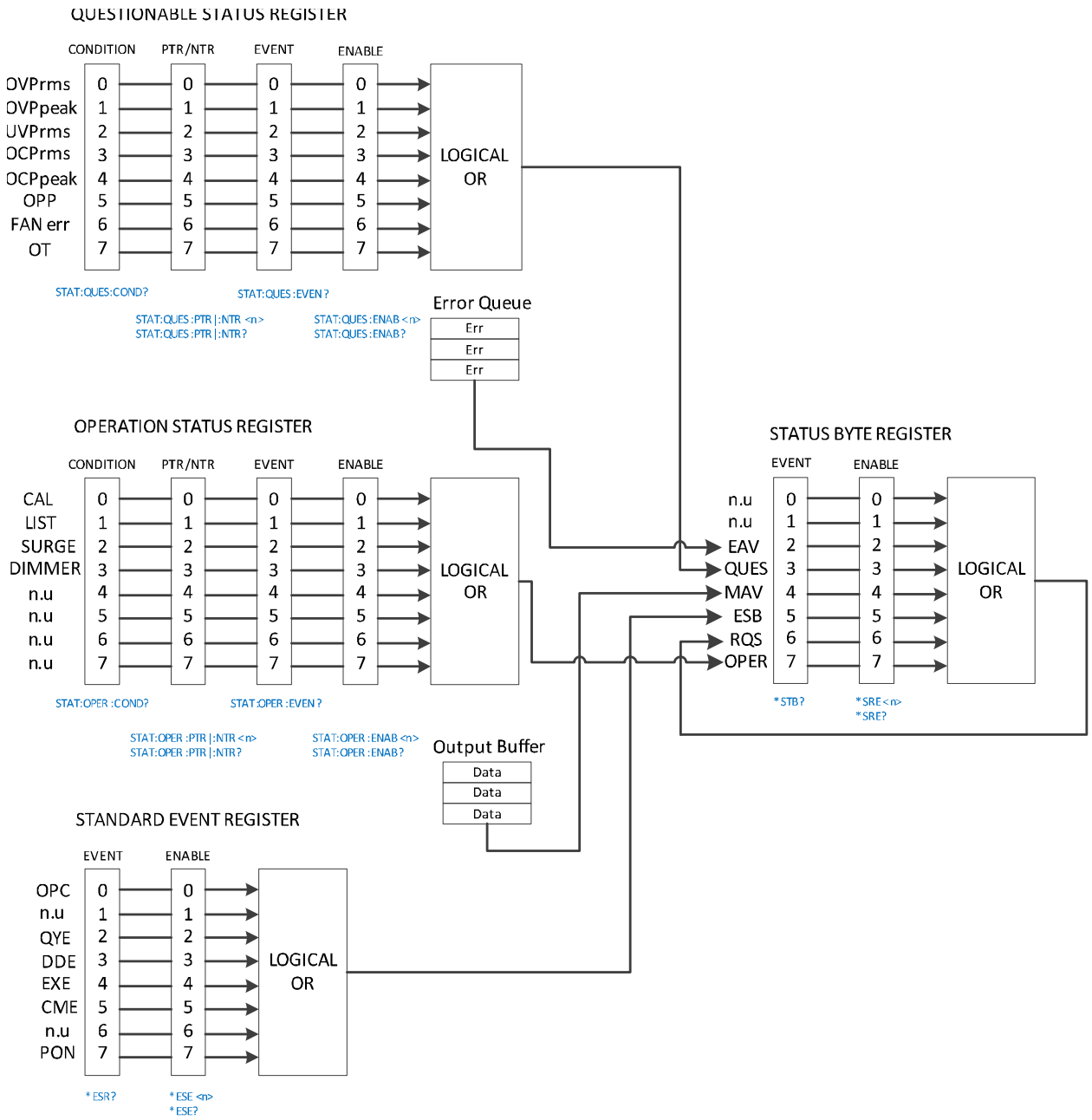
The following table describes the status signals.

BIT	Signal	Meaning
		Operation Status Register
0	CAL	The power supply is under calibration.
1	LIST	The power supply is running the list program.
2	SURGE	The power supply is waiting for a trigger.
3	DIMMER	The power supply is running the surge trap.
		The power supply is running the dimmer.
		Questionable Status Register
0	OVPrms	Output is disabled by the over-voltage protection. RMS
1	OVPpeak	Output is disabled by the over-voltage protection. PEAK
2	UVPrms	Output is disabled by the under-voltage protection. RMS
3	OCPrms	Output is disabled by the over-current protection. RMS
4	OCPpeak	Output is disabled by the over-current protection. PEAK
5	OPP	Output is disabled by the over-power protection.
6	FAN	Fan protection.
7	OT	Output is disabled by the over-temperature protection.
		Standard Event Register
0	OPC	All commands before and including *OPC have been executed.
2	QYE	The instrument tried to read the output buffer but it was empty, a new command line was received before a previous query has been read, or both the input and output buffers are full.
3	DDE	A device-specific error, including a self-test error, calibration error or other device-specific error occurred.
4	EXE	An execution error occurred.
5	CME	A command syntax error occurred.
7	PON	Power has been cycled since the last time the event register was read or

		cleared.
		Status Byte Register
2	EAV	Error buffer available.
3	QUES	This bit is set to 1 when any one status of enabled query status register changes.
4	MAV	Output buffer available.
5	ESB	Bit ESB is set to 1 when the status of an enabled standard event status. Register changes.
6	RQS	If the status of enabled operation register changes, then this bit is set to1.
7	OPER	

2.2 Status Register Structure

The following figure shows the status register structure of the power supply.


QUESTIONABLE STATUS REGISTER
CONDITION
PTR/NTR
EVENT
ENABLE
OVPrms 0
OVPpeak 1
UVPrms 2
OCPrms 3
OCPpeak 4
OPP 5
FAN err 6
OT 7
LOGICAL OR
STAT:QUES:COND?
STAT:QUES:EVEN?
STAT:QUES:PTR|:NTR <n>
STAT:QUES:PTR|:NTR?
STAT:QUES:ENAB <n>
STAT:QUES:ENAB?
Error Queue
Err
Err
Err
OPERATION STATUS REGISTER
CONDITION
PTR/NTR
EVENT
ENABLE
CAL 0
LIST 1
SURGE 2
DIMMER 3
n.u 4
n.u 5
n.u 6
n.u 7
LOGICAL OR
STAT:OPER:COND?
STAT:OPER:EVEN?
STAT:OPER:PTR|:NTR <n>
STAT:OPER:PTR|:NTR?
STAT:OPER:ENAB <n>
STAT:OPER:ENAB?
Output Buffer
Data
Data
Data
STATUS BYTE REGISTER
EVENT
ENABLE
n.u 0
n.u 1
EAV 2
QUES 3
MAV 4
ESB 5
RQS 6
OPER 7
LOGICAL OR
*STB?
*SRE <n>
*SRE?
STANDARD EVENT REGISTER
EVENT
ENABLE
OPC 0
n.u 1
QYE 2
DDE 3
EXE 4
CME 5
n.u 6
PON 7
LOGICAL OR
*ESR?
*ESE <n>
*ESE?

Chapter 3 STATus Subsystem

STATus:QUEStionable[:EVENt]?

Queries the event register for the Questionable Status group. This is a read-only register, which stores (latches) all events that are passed by the Operation NTR and/or PTR filter. Reading the Questionable Status Event register clears it.

Syntax

STATus:QUEStionable[:EVENt]?

Arguments

None

Returns

<NR1>

The value returned is the binary-weighted sum of all bits set in the register.

Related Command

STATus:QUEStionable:ENABle

The following table describes the Questionable Status register bit assignments.

Bit	7	6	5	4	3	2	1	0
Name	OT	FAN err	OPP	OCPpeak	OCPrms	UVPrms	OVPpeak	OVPrms
Decimal Value	128	64	32	16	8	4	2	1

STATus:QUEStionable:CONDition?

Queries the condition register for the Questionable Status group. This is a read only register, which holds the live (unlatched) operational status of the instrument. Reading the Questionable Status Condition register does not clear it.

Syntax

STATus:QUEStionable:CONDition?

Arguments

None

Returns

<NR1>

The value returned is the binary-weighted sum of all bits set in the register.

STATus:QUEStionable:ENABle

Sets the value of the enable register for the Questionable Status group. The enable register is a mask for enabling specific bits from the Operation Event register to set the QUES (questionable summary) bit of the Status Byte register.

Syntax

STATus:QUEStionable:ENABle <NR1>

Arguments

0~255

The value returned is the binary-weighted sum of all bits set in the register.

Default value

See *PSC.

Example

STATus:QUEStionable:ENABle 128

Query Syntax

STATus:QUEStionable:ENABle?

Returns

<NR1>

The value returned is the binary-weighted sum of all bits set in the register.

STATus:QUEStionable:NTRansition

Sets the value of the NTR (Negative-Transition) registers for the Questionable Status group.

Syntax

STATus:QUEStionable:NTRansition <NR1>

Arguments

0~255

Default value

See *PSC.

Example

STATus:QUEStionable:NTRansition 128

Query Syntax

STATus:QUEStionable:NTRansition?

Returns

<NR1>

The value returned is the binary-weighted sum of all bits set in the register.

STATus:QUEStionable:PTRansition

Sets the value of the PTR (Positive-Transition) registers for the Questionable Status group.

Syntax

STATus:QUEStionable:PTRansition <NR1>

Arguments

0~255

Default value

See *PSC.

Example

STATus:QUEStionable:PTRansition 128

Query Syntax

STATus:QUEStionable:PTRansition?

Returns

<NR1>

The value returned is the binary-weighted sum of all bits set in the register.

STATus:OPERation[:EVENt]?

Queries the event register for the Operation Status group. This is a read-only register, which stores (latches) all events that are passed by the Operation NTR and/or PTR filter. Reading the Operation Status Event register clears it.

Query Syntax

STATus:OPERation[:EVENt]?

Arguments

None

Returns

<NR1>

Related Command

STATus:OPERation:ENABle

The following table describes the Operation Status register bit assignments.

Bit	7	6	5	4	3	2	1	0
Name	n.u	n.u	n.u	n.u	DIMMER	SURGE	LIST	CAL
Decimal Value					8	4	2	1

STATus:OPERation:CONDition?

Queries the condition register for the Operation Status group. This is a read-only register, which holds the live (unlatched) operational status of the instrument. Reading the Operation Status Condition register does not clear it.

Query Syntax

STATus:OPERation:CONDition?

Arguments

None

Returns

<NR1>

The value returned is the binary-weighted sum of all bits set in the register.

STATus:OPERation:ENABle

Sets the value of the enable register for the Operation Status group. The enable register is a mask for enabling specific bits from the Operation Event register to set the OPER (operation summary) bit of the Status Byte register.

Syntax

STATus:OPERation:ENABle <NR1>

Arguments

0~255

Default value

See *PSC.

Example

STATus:OPERation:ENABle 128

Query Syntax

STATus:OPERation:ENABle?

Returns

<NR1>

STATus:OPERation:NTRansition

Sets the value of the NTR (Negative-Transition) registers for the Operation Status group.

Syntax

STATus:OPERation:NTRansition <NR1>

Arguments

0~255

Default value

See *PSC.

Example

STATus:OPERation:NTRansition 128

Query Syntax

STATus:OPERation:NTRansition?

Returns

<NR1>

STATus:OPERation:PTRansition

Sets the value of the PTR (Positive-Transition) registers for the Operation Status group.

Syntax

STATus:OPERation:PTRansition <NR1>

Arguments

0~255

Default value

See *PSC.

Example

STATus:OPERation:PTRansition 128

Query Syntax

STATus:OPERation:PTRansition?

Returns

<NR1>

Chapter 4 IEEE-488 Common Commands

IEEE-488 Common commands generally control overall instrument functions, such as reset, status, and synchronization. All common commands consist of a three-letter mnemonic preceded by an asterisk: ***RST *IDN? *SRE 8**.

*CLS

Clear Status Command. Clears the event registers in all register groups. Also clears the status byte and error queue. If *CLS immediately follows a program message terminator (<NL>), then the output queue and the MAV bit are also cleared.

Syntax:

*CLS

Arguments:

None

*ESE

Event status enable command. Sets the value in the enable register for the Standard Event Status group. Each set bit of the register enables a corresponding event. All enabled events are logically ORed into the ESB bit of the status byte.

Syntax

*ESE <NR1>

Arguments

0～255

The value returned is the binary-weighted sum of all bits set in the register.

Default value

See *PSC.

Example

*ESE 128

Query Syntax

*ESE?

Returns

<NR1>

The value returned is the binary-weighted sum of all bits set in the register.

Related Command

*ESR? *PSC *STB?

The following table describes the Standard Event Status register bit assignments.

Bit	7	6	5	4	3	2	1	0
Name	PON	n.u	CME	EXE	DDE	QYE	n.u	OPC
Decimal Value	128		32	16	8	4		1

*ESE?

Query the value of the enable register set by the standard event status group.

Query Syntax

*ESE?

Arguments

None

Returns

<NR1>

The value returned is the binary-weighted sum of all bits set in the register.

*ESR?

Event status event query. Reads and clears the event register for the Standard Event Status group. The event register is a read-only register, which latches all standard events.

Query Syntax

*ESR?

Arguments

None

Returns

<NR1>

The value returned is the binary-weighted sum of all bits set in the register.

Related Command

*CLS *ESE *ESE? *OPC

*IDN?

Identification Query. Returns instrument's identification string, which contains four comma-separated fields. The first field is the manufacturer's name, the second field is the instrument model number, the third field is the serial number, and the fourth field is the firmware revision.

Query Syntax

*IDN?

Arguments

None

Returns

Manufacture, model, serial number, UI ver-DSP1 ver-DSP2 ver-PFC ver-Interface ver

Example

ITECH, M7722, 00000000000004, 1.01-1.00-1.0-1.1-1.2

*OPC

Sets the OPC (operation complete) bit in the standard event register. This occurs at the completion of the pending operation.

Syntax

*OPC

Arguments

None

Query Syntax

*OPC?

Returns

<NR1>

*RST

This command is used to reset the instrument to the factory default value, please refer to Table 4-6 in the User's Manual for details. also avoid executing this command when the instrument is On.

Syntax

*RST

Arguments

None

*SRE

Service request enable command. This sets the value of the Service Request Enable register. This determines which bits from the Status Byte Register are summed to set the Request for Service (RQS) summary bit. A 1 in any Service Request Enable register bit position enables the corresponding Status Byte register bit.

Syntax

*SRE <NR1>

Arguments

0～255

Default value

See *PSC.

Example

*SRE 128

Query Syntax

*SRE?

Returns

<NR1>

Related Command

*ESE *ESR? *PSC *STB?

*STB?

Status byte query. The Status Byte is a read-only register and the bits are not cleared when it is read.

Query Syntax

*STB?

Arguments

None

Returns

<NR1>

Related Command

*CLS *ESE *ESR

The following table describes the Status Byte Register register bit assignments.

Bit	7	6	5	4	3	2	1	0
Name	OPER	RQS	ESB	MAV	QUES	EAV	n.u	n.u
Decimal Value	128	64	32	16	8	4		

*PSC

This command controls whether a service request will generate when the power supply is re-generalized.

- 1 OR ON: when the power supply is generated, the values in the Status Byte Enable register, Operational Event Enable register, Query Event Enable register and Standard Event Enable register are cleared.
- 0 OR OFF: the values in the Status Byte Enable register, Operational Event Enable register, Query Event Enable register and Standard Event Enable register are saved in NVM, which will be used at regenerating.

Syntax

*PSC <ON|1|OFF|0>

Arguments

ON|1|OFF|0

Query Syntax

*PSC?

Returns

<ON|OFF>

*SAV

Save the instrument several parameter settings to 10 nonvolatile memories, and the position can be set from 0 to 9. When shipped, locations 0 through 9 are empty. The specific parameter information can be found in the user manual.

Syntax

*SAV <NR1>

Arguments

0~9

*RCL

Recalls a saved instrument state. This restores the instrument to a state that was previously stored in locations 0 through 9 with the *SAV command.

Syntax

*RCL <NR1>

Arguments

0～9

Chapter 5 SYSTem Subsystem

SYSTem:ERRor?

This command is used to query the error information of the instrument. When the ERROR indicator on the front panel is lit, it indicates that one or more errors have occurred in the hardware or command syntax of the detected instrument. Up to 20 sets of error messages can be stored in the error queue. This command is sent once to read an error message from the error queue.

- The front-panel ERROR annunciator turns on when one or more errors are currently stored in the error queue. Error retrieval is first-in-first-out (FIFO), and errors are cleared as you read them. When you have read all errors from the error queue, the ERROR annunciator turns off.
- If more than 20 errors have occurred, the last error stored in the queue (the most recent error) is replaced with -350 ("Error queue overflow"). No additional errors are stored until you remove errors from the queue. If no errors have occurred when you read the error queue, the instrument responds with +0 ("No error").
- If the instrument is turned off or the *CLS (clear status) command is sent, the error message in the error queue will be cleared. The *RST command will not clear the error message in the error queue.

The error code and error information are as follows.

(0) No error
(-102) Syntax error
(-103) Invalid separator
(-108) Parameter not allowed
(-109) Missing parameter
(-113) Undefined header
(-131) Invalid suffix
(-138) Suffix not allowed
(-200) Execution Error
(-222) Data out of Range
(-350) Queue overflow

SYSTem:CLEar

This command is used to clear the error queue.

Syntax:

SYSTem:CLEar

Arguments:

None

Returns:

None

SYSTem:REMote

This command is used to set the instrument to the remote control mode via the communication interface. Except the **[Shift]+[Link](** Local**)** key and **[On/Off]** key on the front panel, other keys are locked and cannot be used.

Syntax:

SYST:REM

Arguments:

None

Query Syntax:

None

SYSTem:LOCal

This command is used to set the instrument to local mode, i.e. panel control mode. All keys on the front panel will be available after executing this command.

Syntax

SYST:LOC

Arguments:

None

Query Syntax:

None

SYSTem:RWLock

This command is transmit through interface, the same function as SYST:REM, but the difference is: this command can lock “LOCAL” button as well. When this command is executed, the “Local” button can’t switch to local control mode.

Syntax

SYST:RWL

Arguments

None

SYSTem:BEEPer

This command is used to turn on/turn off the beeper, when the arguments is set to 1/ON, the beeper is enabled and press keyboard will beeper, otherwise, it is mute.

Syntax

SYSTem:BEEPer <OFF|ON|0|1>

Arguments

<OFF|ON|0|1>

Example

SYST:BEEP 1

SYSTem:BEEPer?

This command is used to query the status of the buzzer.

Syntax

SYSTem:BEEPer?

Arguments

None

Example

SYST:BEEP?

Returns:

<ON|OFF>

SYSTem:POWnon

This command controls the settings and output state when the power supply is powered on.

Syntax

SYSTem:POWnon <LAST+OFF|LAST|RESET|0|1|2>

Arguments

<LAST+OFF|LAST|RESET|0|1|2>

Example

SYSTem:POWnon LAST+OFF

SYSTem:POWnon?

This command is used to query some parameter settings or working status when the instrument is powered on.

Syntax

SYSTem:POWnon?

Arguments:

None

Returns:

<LAST+OFF|LAST|RESET>

SYSTem:COMMunicate:SELect

This command is used to select the communication interface.

Syntax

SYSTem:COMMunicate:SELect <RS232|GPIB|USB|LAN|CAN|ANALOG|RS485>

Arguments

<RS232|GPIB|USB|LAN|CAN|ANALOG|RS485|0|1|2|3|4|5|6>

- When inserting the machine expansion slot into the IT-E1205 optional card,

the parameter is <GPIB|1>。

- When inserting the machine expansion slot into the IT-E1206 optional card, the parameter is <USB|LAN|2|3>.
- When inserting the machine expansion slot into the IT-E1207 optional card, the parameter is <RS232|CAN|0|4>.
- When inserting the machine expansion slot into the IT-E1208 optional card, the parameter is <ANALOG|RS485|5|6>.
- When inserting the machine expansion slot into the IT-E1209 optional card, the parameter is <USB|2>.

Example

SYST:COMM:SEL RS232

Query Syntax

SYSTem:COMMunicate:SELect?

Returns

<RS232|GPIB|USB|LAN|CAN|ANALOG|RS485>

SYSTem:COMMunicate:SERial[:SELF]:BAUDrate

This command is used to set the baudrate of RS232.

Syntax

SYSTem:COMMunicate:SERial[:SELF]:BAUDrate <4800|9600|19200|38400|57600|115200>

Arguments

<4800|9600|19200|38400|57600|115200>

Query Syntax

SYSTem:COMMunicate:SERial[:SELF]:BAUDrate?

Returns

<4800|9600|19200|38400|57600|115200>

SYSTem:COMMunicate:GPIB[:SELF]:ADDRess

This command is used to set the GPIB communication address.

Syntax

SYSTem:COMMunicate:GPIB[:SELF]:ADDRess <NR1>

Arguments:

<NR1>

Setting range: 1 to 30.

Query Syntax:

SYSTem:COMMunicate:GPIB[:SELF]:ADDRess?

Returns:

<NR1>

SYSTem:COMMunicate:CAN[:SELF]:ADDRess

This command is used to set the CAN communication address.

Syntax

SYSTem:COMMunicate:CAN[:SELF]:ADDRess <NR1>

Arguments:

<NR1>

Setting range: 1 to 127.

Query Syntax:

SYSTem:COMMunicate:CAN[:SELF]:ADDRess?

Returns:

<NR1>

Note

The IT7723P power supply does not support CAN communication, so this command is not applicable to the IT7723P.

SYSTem:COMMunicate:CAN[:SELF]:BAUDrate

This command is used to set the baudrate of CAN.

Syntax

SYSTem:COMMunicate:CAN[:SELF]:BAUDrate <0|1|2>

Arguments:

<0|1|2>

- 0: Indicates that the CANBUS baud rate of the power supply is 500K.
- 1: Indicates that the CANBUS baud rate of the power supply is 250K.
- 2: Indicates that the CANBUS baud rate of the power supply is 125K.

Query Syntax:

SYSTem:COMMunicate:CAN[:SELF]:BAUDrate?

Returns:

<0|1|2>

Note

The IT7723P power supply does not support CAN communication, so this command is not applicable to the IT7723P.

SYSTem:COMMunicate:LAN:IPCONFig

This command is used to set the IP mode of the LAN port.

- MANU: The user manually sets the IP related parameters.
- AUTO: The system automatically configures IP related parameters.

Syntax

SYSTem:COMMunicate:LAN:IPCONFig <0|1|AUTO|MANU>

Arguments

<0|1|AUTO|MANU>

Query Syntax

SYSTem:COMMunicate:LAN:IPCONFig?

Returns

<AUTO|MANU>

SYSTem:COMMunicate:LAN:CURRent:ADDRess?

This command is used to query the IP address of the instrument.

Syntax

SYSTem:COMMunicate:LAN:CURRent:ADDRess?

Returns:

<String>

Example: 192.168.50.123

SYSTem:COMMunicate:LAN:CURRent:SMASk?

This command is used to query the subnet mask of the LAN communication.

Syntax

SYSTem:COMMunicate:LAN:CURRent:SMASk?

Returns:

<String>

Example: 255.255.255.0

SYSTem:COMMunicate:LAN:CURRent:DGATeway?

This command is used to query the gateway address of the LAN communication.

Syntax

SYSTem:COMMunicate:LAN:CURRent:DGATeway?

Returns:

<String>

Example: 192.168.50.255.

SYSTem:COMMunicate:LAN:CURRent:DNS1?

This command is used to query the DNS primary address of the LAN communication.

Syntax

SYSTem:COMMunicate:LAN:CURRent:DNS1?

Returns:

<String>

SYSTem:COMMunicate:LAN:CURRent:DNS2?

This command is used to query the DNS secondary address of the LAN communication.

Syntax

SYSTem:COMMunicate:LAN:CURRent:DNS2?

Returns:

<String>

SYSTem:COMMunicate:LAN:ADDRess

This command is used to set the IP address of the instrument.

Syntax

SYSTem:COMMunicate:LAN:ADDRess <String>

Arguments:

<String>

Example: "192.168.50.123"

Query Syntax:

SYSTem:COMMunicate:LAN:ADDRess?

Returns:

<String>

SYSTem:COMMunicate:LAN:SMASk

This command is used to set the subnet mask.

Syntax

SYSTem:COMMunicate:LAN:SMASk <String>

Arguments:

<String>

Example: "255.255.255.0"

Query Syntax:

SYSTem:COMMunicate:LAN:SMASk?

Returns:

<String>

SYSTem:COMMunicate:LAN:DGATeway

This command is used to set the gateway address of the LAN communication.

Syntax

SYSTem:COMMunicate:LAN:DGATeway <String>

Arguments

<String>

Example: "192.168.50.255"

Query Syntax

SYSTem:COMMunicate:LAN:DGATeway?

Returns

<String>

SYSTem:COMMunicate:LAN:DNS1

This command sets DNS primary address for LAN.

Syntax

SYSTem:COMMunicate:LAN:DNS1 <String>

Arguments:

<String>

Example: "192.168.0.1"

Query Syntax:

SYSTem:COMMunicate:LAN:DNS1?

Returns:

<String>

SYSTem:COMMunicate:LAN:DNS2

This command sets DNS secondary address for LAN.

Syntax

SYSTem:COMMunicate:LAN:DNS2 <String>

Arguments

<String>

Example: "192.168.0.2"

Query Syntax

SYSTem:COMMunicate:LAN:DNS2?

Returns:

<String>

SYSTem:COMMunicate:LAN:MACaddress

This command sets MAC address for LAN.

Syntax

SYSTem:COMMunicate:LAN:MACaddress <String>

Arguments:

<String>

Example: "05:04:03:02:01:00"

Query Syntax:

SYSTem:COMMunicate:LAN:MACaddress?

Returns:

<String>

SYSTem:COMMunicate:LAN:PING[:STATe]

This command enables or disables the Ping function.

Syntax

SYSTem:COMMunicate:LAN:PING[:STATe] <0|1|OFF|ON>

Arguments:

<0|1|OFF|ON>

Query Syntax:

SYSTem:COMMunicate:LAN:PING[:STATe]?

Returns:

<OFF|ON>

SYSTem:COMMunicate:LAN:MDNS[:STATe]

This command enables or disables the MDNS Server function.

Syntax

SYSTem:COMMunicate:LAN:MDNS[:STATe] <0|1|OFF|ON>

Arguments:

<0|1|OFF|ON>

Query Syntax:

SYSTem:COMMunicate:LAN:MDNS[:STATe]?

Returns:

<OFF|ON>

SYSTem:COMMunicate:LAN:HTTP[:STATe]

This command enables or disables the HTTP Server function.

Syntax

SYSTem:COMMunicate:LAN:HTTP[:STATe] <0|1|OFF|ON>

Arguments:

<0|1|OFF|ON>

Query Syntax:

SYSTem:COMMunicate:LAN:HTTP[:STATe]?

Returns:

<OFF|ON>

SYSTem:COMMunicate:LAN:VXI11[:STATe]

This command enables or disables the VXI-11 Server function.

Syntax

SYSTem:COMMunicate:LAN:VXI11[:STATe] <0|1|OFF|ON>

Arguments:

<0|1|OFF|ON>

Query Syntax:

SYSTem:COMMunicate:LAN:VXI11[:STATe]?

Returns:

<OFF|ON>

SYSTem:COMMunicate:LAN:SOCket[:STATe]

This command enables or disables the Socket Server function.

Syntax

SYSTem:COMMunicate:LAN:SOCket[:STATe] <0|1|OFF|ON>

Arguments:

<0|1|OFF|ON>

Query Syntax:

SYSTem:COMMunicate:LAN:SOCket[:STATe]?

Returns:

<OFF|ON>

SYSTem:COMMunicate:LAN:TELnet[:STATe]

This command enables or disables the Telnet Server function.

Syntax

SYSTem:COMMunicate:LAN:TELnet[:STATe] <0|1|OFF|ON>

Arguments:

<0|1|OFF|ON>

Query Syntax:

SYSTem:COMMunicate:LAN:TELnet[:STATe]?

Returns:

<OFF|ON>

SYSTem:COMMunication:LAN:SOCket:PORT

This command sets the socket port for the LAN communication.

Syntax

SYSTem:COMMunication:LAN:SOCket:PORT <NR1>

Arguments:

<NR1> (100 ~ 65535)

Query Syntax:

SYSTem:COMMunication:LAN:SOCket:PORT?

Returns:

< NR1>

SYSTem:COMMunicate:LAN:CONFig:HOSTname

This command sets the LXI Host name for the LAN communication.

Syntax

SYSTem:COMMunicate:LAN:CONFig:HOSTname <String>

Arguments:

<String>

Example: "IT-M7722"

Query Syntax:

SYSTem:COMMunicate:LAN:CONFig:HOSTname?

Returns:

< String>

SYSTem:COMMunicate:LAN:CONFig:DESCriptionname

This command sets the LXI Description name for the LAN communication.

Syntax

SYSTem:COMMunicate:LAN:CONFig:DESCriptionname <String>

Arguments:

<String>

Example: "IT-M7723-ACPowerSupply"

Query Syntax

SYSTem:COMMunicate:LAN:CONFig:DESCriptionname?

Returns:

< String>

SYSTem:COMMunicate:LAN:INFormation:HOSTname?

This command is used to query the LXI Host name of the LAN communication.

Syntax

SYSTem:COMMunicate:LAN:INFormation:HOSTname?

Returns:

< String>

SYSTem:COMMunicate:LAN:INFormation:DESCription?

This command is used to query the LXI Description name of the LAN communication.

Syntax

SYSTem:COMMunicate:LAN:INFormation:DESCription?

Returns:

< String>

SYSTem:COMMunicate:LAN:IPMODe?

This command is used to query the LXI Config IP status of the LAN communication.

Syntax

SYSTem:COMMunicate:LAN:IPMODe?

Returns:

< String>

Example:

"Search DHCP Server..."

"DHCP"

"Search Auto-IP"

"Auto-IP"

"Manual"

SYSTem:COMMunicate:LAN:RESET

This command makes the LAN settings valid.

Syntax

SYSTem:COMMunicate:LAN:RESET

Arguments

None

SYSTem:COMMunicate:485[:SELF]:BAUDrate

This command is used to set the baudrate of RS485.

Syntax

SYSTem:COMMunicate:SERial[:SELF]:BAUDrate <4800|9600|19200|38400|57600|115200>

Arguments

<4800|9600|19200|38400|57600|115200>

Query Syntax

SYSTem:COMMunicate:SERial[:SELF]:BAUDrate?

Returns

<4800|9600|19200|38400|57600|115200>

SYSTem:COMMunicate:485[:SELF]:ADDRess

This command is used to set the RS485 communication address.

Syntax

SYSTem:COMMunicate:485[:SELF]:ADDRess <NR1>

Arguments

<NR1>

Settable range: 1 to 126.

Query Syntax

SYSTem:COMMunicate:485[:SELF]:ADDRess?

Returns

<NR1>

Chapter 6 SOURce Subsystem

[SOURce:]RELAy:MODe

This command is used to set the relay mode.

Syntax

[SOURce:]RELAy:MODe <0|1|OUTSYN|NC>

Arguments:

<0|1|OUTSYN|NC>

OUTSYN: The relay is linked with Output; NC:The relay is normally closed. Whether

Query Syntax:

[SOURce:]RELAy:MODe?

Returns:

<OUTSYN|NC>

[SOURce:]NORMal:VOLTage:AC[:LEVel][:IMMediate][:AMPLitude]

This command sets the AC voltage of the power supply.

Syntax

[SOURce:]NORMal:VOLTage:AC[:LEVel][:IMMediate][:AMPLitude] <NRf>

Arguments:

<NRf>

Query Syntax:

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

Returns:

<NRf>

[SOURce:]NORMal:VOLTage:DC[:LEVel][:IMMediate]

This command sets the DC voltage of the power supply.

Syntax

[SOURce:]NORMal:VOLTage:DC[:LEVel][:IMMediate] <NRf>

Arguments:

<NRf>

Query Syntax:

[SOURce:]NORMal:VOLTage:DC[:LEVel][:IMMediate]?

Returns:

<NRf>

[SOURce:]NORMal:FREQuency[:LEVel][:IMMediate]

This command sets the AC frequency of the power supply.

Syntax

[SOURce:]NORMal:FREQuency[:LEVel][:IMMediate] <NRf>

Arguments:

<NRf>

Query Syntax:

[SOURce:]NORMal:FREQuency[:LEVel][:IMMediate]?

Returns:

<NRf>

[SOURce:]NORMal:VOLTage:AC:MAX[:LEVel]

This command sets the maximum AC voltage of the power supply.

Syntax

[SOURce:]NORMal:VOLTage:AC:MAX[:LEVel] <NRf>

Arguments:

<NRf>

Query Syntax:

[SOURce:]NORMal:VOLTage:AC:MAX[:LEVel]?

Returns:

<NRf>

[SOURce:]NORMal:VOLTage:AC:MIN[:LEVel]

This command sets the minimum AC voltage of the power supply.

Syntax

[SOURce:]NORMal:VOLTage:AC:MIN[:LEVel] <NRf>

Arguments:

<NRf>

Query Syntax:

[SOURce:]NORMal:VOLTage:AC:MIN[:LEVel]?

Returns:

<NRf>

[SOURce:]NORMal:VOLTage:DC:MAX[:LEVel]

This command sets the maximum DC voltage of the power supply.

Syntax

[SOURce:]NORMal:VOLTage:DC:MAX[:LEVel] <NRf>

Arguments:

<NRf>

Query Syntax:

[SOURce:]NORMal:VOLTage:DC:MAX[:LEVel]?

Returns:

<NRf>

[SOURce:]NORMal:VOLTage:DC:MIN[:LEVel]

This command sets the minimum DC voltage of the power supply.

Syntax

[SOURce:]NORMal:VOLTage:DC:MIN[:LEVel] <NRf>

Arguments:

<NRf>

Query Syntax:

[SOURce:]NORMal:VOLTage:DC:MIN[:LEVel]?

Returns:

<NRf>

[SOURce:]NORMal:FREQuency:MAX[:LEVel]

This command sets the maximum AC frequency of the power supply.

Syntax

[SOURce:]NORMal:FREQuency:MAX[:LEVel] <NRf>

Arguments:

<NRf>

Query Syntax:

[SOURce:]NORMal:FREQuency:MAX[:LEVel]?

Returns:

<NRf>

[SOURce:]NORMal:FREQuency:MIN[:LEVel]

This command sets the minimum AC frequency of the power supply.

Syntax

[SOURce:]NORMal:FREQuency:MIN[:LEVel] <NRf>

Arguments:

<NRf>

Query Syntax:

[SOURce:]NORMal:FREQuency:MIN[:LEVel]?

Returns:

<NRf>

[SOURce:]NORMal:PHASe:STARt[:LEVel][:IMMediate]

This command sets the AC start phase angle of the power supply.

Syntax

[SOURce:]NORMal:PHASe:STARt[:LEVel][:IMMediate] <NRf>

Arguments:

<NRf>

Query Syntax:

[SOURce:]NORMal:PHASe:STARt[:LEVel][:IMMediate]?

Returns:

<NRf>

[SOURce:]NORMal:PHASe:STOP[:LEVel][:IMMediate]

This command sets the AC stop phase angle of the power supply.

Syntax

[SOURce:]NORMal:PHASe:STOP[:LEVel][:IMMediate] <NRf>

Arguments:

<NRf>

Query Syntax:

[SOURce:]NORMal:PHASe:STOP[:LEVel][:IMMediate]?

Returns:

<NRf>

[SOURce:]NORMal:MODE

This command sets the output mode of the power supply.

Syntax

[SOURce:]NORMal:MODE <AC|DC|AC+DC>

Arguments:

<AC|DC|AC+DC>

Query Syntax:

[SOURce:]NORMal:MODE?

Returns:

<AC|DC|AC+DC>

[SOURce:]OUTPut[:STATe]

This command sets the output state of the power supply.

Syntax

[SOURce:]OUTPut[:STATe] <0|1|OFF|ON>

Arguments:

<0|1|OFF|ON>

Query Syntax:

[SOURce:]OUTPut[:STATe]?

Returns:

<OFF|ON>

[SOURce:]NORMal:VRISETIME

This command sets the DC voltage rising time of the instrument.

Syntax

[SOURce:]NORMal:VRISETIME <NR1>

Arguments:

<NR1> (1~99999)ms

Query Syntax:

[SOURce:]NORMal:VRISETIME?

Returns:

<NR1>

[SOURce:]NORMal:VRISETIME:AC

This command sets the AC voltage rising time of the instrument.

Note: IT-M7723P does not support this function.

Syntax

[SOURce:]NORMal:VRISETIME:AC <NR1>

Arguments:

<NR1> (1~99999)ms

Query Syntax:

[SOURce:]NORMal:VRISETIME:AC?

Returns:

<NR1>

Chapter 7 SOURce Subsystem(IT-M7700)

The commands in this chapter is a special command of IT-M7721/IT-M7722/ IT-M7723/IT-M7723E/IT-M7722D/IT-M7722E/IT-M7723D/IT-M7723P, but it's not suitable for IT-M7721L/IT-M7722L.

[SOURce:]NORMal:CURRent:RANGe

This command is used to set the current range of the power supply.

Syntax

[SOURce:]NORMal:CURRent:RANGe <0|1|2|AUTO|HIGH|LOW>

Arguments:

<0|1|2|AUTO|HIGH|LOW>

Query Syntax:

[SOURce:]NORMal:CURRent:RANGe?

Returns:

<AUTO|HIGH|LOW>

Note

This command is a special command of IT-M7721/IT-M7722/IT-M7723E/IT-M7722D/ IT-M7722E/IT-M7723D.

[SOURce:]NORMal:VOLTage:RANGe

This command is used to set the voltage range of the power supply.

Syntax

[SOURce:]NORMal:VOLTage:RANGe <0|1|LOW|HIGH>

Arguments

<0|1|LOW|HIGH>

Query Syntax

[SOURce:]NORMal:VOLTage:RANGe?

Returns

<0|1|LOW|HIGH>

Note

This command is a special command of IT-M7723.

The response time of this command is at least greater than 100ms.

[SOURce:]NORMal:WAVE

This command is used to set the wave.

Syntax

[SOURce:]NORMal:WAVE
<0|1|2|3|4|SINE|SQUA|TRIANGLE|SAW|CLIPSINE>

Arguments:

<0|1|2|3|4|SINE|SQUA|TRIANGLE|SAW|CLIPSINE>

Query Syntax:

[SOURce:]NORMal:WAVE?

Returns:

<SINE|SQUA|TRIANGLE|SAW|CLIPSINE>

[SOURce:]NORMal:WAVE:THD

This command is used to set the THD.

Syntax

[SOURce:]NORMal:WAVE:THD <NR1>

Arguments:

<NR1> 1~30 => THD1~THD30

Query Syntax:

[SOURce:]NORMal:WAVE:THD?

Returns:

<String> THD1~THD30

[SOURce:]NORMal:WAVE:USER

This command is used to set the USER wave.

Syntax

[SOURce:]NORMal:WAVE:USER <NR1>

Arguments:

<NR1> 1~5 => USER1~USER5

Query Syntax:

[SOURce:]NORMal:WAVE:USER?

Returns:

<String> USER1~USER5

[SOURce:]NORMal:WAVE:CSINe

This command is used to set the clipped percentage of the Clip Sine waveform.

Syntax

[SOURce:]NORMal:WAVE:CSINe <NRf>

Arguments:

<NRf> 0 ~ 100.0 => (0~100.0%)

Query Syntax:

[SOURce:]NORMal:WAVE:CSINe?

Returns:

<NRf>

[SOURce:]NORMal:DIMMer[:PHASe]

This command is used to set the phase angle of the phase dimming function.

Syntax

[SOURce:]NORMal:DIMMer[:PHASe] <NR1>

Arguments:

<NRf> (0.0° ~ 180.0°)

Query Syntax:

[SOURce:]NORMal:DIMMer[:PHASe]?

Returns:

<NRf>

[SOURce:]NORMal:DIMMer:MODe

The command is used to configure the phase dimming mode: leading/trailing edge dimming.

Syntax

[SOURce:]NORMal:DIMMer:MODe <0|1|2|OFF|LEADING|TRAILING>

Arguments:

<0|1|2|OFF|LEADING|TRAILING>

Query Syntax:

[SOURce:]NORMal:DIMMer:MODe?

Returns:

<OFF|LEADING|TRAILING>

[SOURce:]NORMal:SURGETRAP:MODe

This command enables or disables the Surge/Trap Function.

Syntax

[SOURce:]NORMal:SURGETRAP:MODe <0|1|OFF|ON>

Arguments:

<0|1|OFF|ON>

Query Syntax:

[SOURce:]NORMal:SURGETRAP:MODe?

Returns:

<OFF|ON>

[SOURce:]NORMal:SURGETRAP:PERIOD

The command is used to set the period of the surge/trap.

Syntax

[SOURce:]NORMal:SURGETRAP:PERIOD <NR1>

Arguments

<NR1> 1 ~ 999 (cycle)

Query Syntax:

[SOURce:]NORMal:SURGETRAP:PERIOD?

Returns:

<NR1>

[SOURce:]NORMal:SURGETRAP:WIDTH

The command is used to set the surge/trap width.

Syntax

[SOURce:]NORMal:SURGETRAP:WIDTH <NRf>

Arguments:

<NRf> 0.01 ~ 100.00mS

Query Syntax:

[SOURce:]NORMal:SURGETRAP:WIDTH?

Returns:

<NRf>

[SOURce:]NORMal:SURGETRAP:PERCENT

The command is used to set the percentage of the surge/trap amplitude to AC signal amplitude(rms).

Syntax

[SOURce:]NORMal:SURGETRAP:PERCENT <NRf>

Arguments:

<NRf> 0 ~ 500.0%

Query Syntax:

[SOURce:]NORMal:SURGETRAP:PERCENT?

Returns:

<NRf>

[SOURce:]NORMal:OFFTIMER:MODe

This command is used to set the output timer function.

Syntax

[SOURce:]NORMal:OFFTIMER:MODe <0|1|OFF|ON>

Arguments

<0|1|OFF|ON>

Query Syntax:

[SOURce:]NORMal:OFFTIMER:MODe?

Returns:

<OFF|ON>

[SOURce:]NORMal:OFFTIMER

This command is used to set the setting value of output timer (unit: ms).

Syntax

[SOURce:]NORMal:OFFTIMER <NR1>

Arguments

<NR1> 10~10000000

Query Syntax

[SOURce:]NORMal:OFFTIMER?

Returns

<NR1>

Chapter 8 PROTect Subsystem

PROTect:RMS:VOLTage

This command is used to set the OVPrms value.

Syntax

PROTect:RMS:VOLTage <NRf>

Arguments:

<NRf>

Query Syntax:

PROTect:RMS:VOLTage?

Returns:

<NRf>

PROTect:PEAK:VOLTage

This command is used to set the OVPpeak value.

Syntax

PROTect:PEAK:VOLTage <NRf>

Arguments:

<NRf>

Query Syntax:

PROTect:PEAK:VOLTage?

Returns:

<NRf>

PROTect:RMS:UNVOLTage

This command is used to set the UVPrms value.

Syntax

PROTect:RMS:UNVOLTage <NRf>

Arguments:

<NRf>

Query Syntax:

PROTect:RMS:UNVOLTage?

Returns:

<NRf>

PROTect:RMS:CURRent

This command is used to set the OCPrms value.

Syntax

PROTect:RMS:CURRent <NRf>

Arguments:

<NRf>

Query Syntax:

PROTect:RMS:CURRent?

Returns:

<NRf>

PROTect:PEAK:CURRent

This command is used to set the OCPpeak value.

Syntax

PROTect:PEAK:CURRent <NRf>

Arguments:

<NRf>

Query Syntax:

PROTect:PEAK:CURRent?

Returns:

<NRf>

PROTect:RMS:CURRent:TIME

This command is used to set the delay time of OCP.

Syntax

PROTect:RMS:CURRent:TIME <NR1>

Arguments:

<NR1> 0 ~ 9999

Query Syntax:

PROTect:RMS:CURRent:TIME?

Returns:

<NR1>

PROTect:RMS:CURRent:MAX[:LEVel]

This command is used to set the maximum value of the OCPrms setting value.

Syntax

PROTect:RMS:CURRent:MAX[:LEVel] <NRf>

Arguments:

<NRf>

Query Syntax:

PROTect:RMS:CURRent:MAX[:LEVel]?

Returns:

<NRf>

PROTect:RMS:CURRent:MIN[:LEVel]

This command is used to set the minimum value of the OCPrms setting value.

Syntax

PROTect:RMS:CURRent:MIN[:LEVel] <NRf>

Arguments:

<NRf>

Query Syntax:

PROTect:RMS:CURRent:MIN[:LEVel]?

Returns:

<NRf>

PROTect:CLEar

This command clears the protection status.

Syntax

PROTect:CLEar

Arguments:

None

PROTect:SENse:CHECK

This command is used to set the sense check protection.

Syntax

PROTect:SENse:CHECK <0|1|OFF|ON>

Arguments:

<0|1|OFF|ON>

Query Syntax:

PROTect:SENse:CHECK?

Returns:

<OFF|ON>

PROTect:POWer

This command sets the over-power limit of the power supply.

Syntax

PROTect:POWer <NRf>

Arguments:

<NRf>

Query Syntax:

PROTect:POWer?

Returns:

<NRf>

PROTect:MAX:CURRent:LIMit

This command sets the max current limit of the power supply.

Syntax

PROTect:MAX:CURRent:LIMit <NRf>

Arguments:

<NRf>

Query Syntax:

PROTect:MAX:CURRent:LIMit?

Returns:

<NRf>

Chapter 9 LIST Subsystem(IT-M7700)

LIST commands is a special command of IT-M7721/IT-M7722/IT-M7723/ IT-M7723E/IT-M7722D/IT-M7722E/IT-M7723D/IT-M7723P, but it's not suitable for IT-M7721L/IT-M7722L.

LIST:TRIGger:MODe

This command is used to set the trigger mode to run the List.

Syntax

LIST:TRIGger:MODe <OFF|ON|0|1>

Arguments:

<OFF|ON|0|1>

- Off: After the Trigger key is pressed, the instrument will automatically run the real-time List file.
- On: After the Trigger key is pressed, the instrument will run the realtime List file in single step.

Query Syntax:

LIST:TRIGger:MODe?

Returns:

<OFF|ON>

LIST:STATe

This command is used to set the mode of recall list.

Syntax

LIST:STATe <OFF|0-5>

Arguments:

<OFF|0-5>

- 1 - 5: indicate that the List Mode function is enabled, and LIST1 to LIST5 file is called.
- Off: indicating that the List Mode function is disabled.

- 0 also indicates OFF.

Query Syntax:

LIST:STATe?

Returns:

<OFF|1-5>

LIST:RECOrd:NUM

This command is used to set the List step number.

Syntax

LIST:RECOrd:NUM <NR1>

Arguments:

<1~50>

Query Syntax:

LIST:RECOrd:NUM?

Returns:

<1~50>

LIST:REPeat

This command is used to set the running times of list file.

Syntax

LIST:REPeat <NR1>

Arguments:

1~50000

Query Syntax:

LIST:REPeat?

Returns:

<NR1>

LIST:ENDState

This command is used to set end of List.

Syntax

LIST:ENDState <0|1>

Arguments:

<0|1>: 0 Stop output, 1 Keep last state.

Query Syntax:

LIST:ENDState?

Returns:

<0|1>

LIST:RUN:RECOrd?

This command is used to query the record index during List running.

Syntax

LIST:RUN:RECOrd?

Arguments:

None

Returns:

<NR1>

LIST:RUN:STATe?

This command queries the List operation state.

Syntax

LIST:RUN:STATe?

Arguments:

None

Returns:

<0|1>

LIST:CONFigure

This command is used to set the Arguments for each group of Lists. Configure the Arguments format as: "List total steps, Jump, Repeat, End State, and retain Arguments". Detailed Arguments are explained below.

Arguments	Descriptions
List total steps	Set the total steps of the real-time list program: 1-50.
Jump	Set the step where the real-time List starts execution: 1-50.
Repeat	Set the number of list repetitions within the range from 1 to 50000.
End State	Set the running state after the list program is running over. ● 0: the instrument output is turned off after the execution is completed; ● 1: keep outputting the last step when the dwell time is over.
Retain Arguments	This value is 0.
V Range(IT-M7723)	0: Low range, 1: High range.

Syntax

LIST:CONFigure <String>

Arguments:

<String>

Example

LIST:CONFigure "50,1,100,0,0,0" (IT-M7723)

LIST:CONFigure "50,1,100,0,0" (Other models except 7723.)

Query Syntax:

LIST:CONFigure?

Returns:

<String>

LIST:RECOrder

This command is used to set the record Arguments of a single Step in the List. Record the Arguments format as: Step index value, "Type, Wave Type, Level, Vac, Vdc, Freq, Time, Start Phase, Stop Phase, Kac, Kdc, Kfreq, Repeat". Detailed Arguments are explained below.

Arguments	Description
Step index value	Set the current Step number.
Type	Set the single step type: 0:Time.
Wave Type	Set the single-step waveform type: 0~39. ● 0:Sine ● 1:Square ● 2:Triangle ● 3:Saw ● 4:Clip Sine ● 5~34:THD1~THD30 ● 35~39:USER1~USER5
Level	Set the clipping percentage. This option is only valid when the Clip Sine output waveform is selected.
Vac	The AC voltage value set in a single step.
Vdc	The DC voltage value set in a single step
Freq	The frequency value set in a single step.
Time	Dwell time for single step: 0~100000000ms.
Start Phase	Start phase angle of single-step waveform.
Stop Phase	Stop phase angle of single-step waveform.
Kac	AC rising slope of single step.
Kdc	DC rising slope of single step.
Kfreq	Frequency rising slope of single step.
Repeat	Set the number of repeats for the step: 1-50,000.

NOTE

The minimum unit of Kac, Kdc, Kfreq is mS (millisecond), these three parameters are the parameters of the unit time variation. When these three parameters are set, the decimal point will be automatically rounded off. When these three parameters are 0, it represents the fastest change.

Syntax

LIST:RECOrder <NR1>,<String>

Arguments:

<NR1>,<String>

- <NR1>: Edit index number of Record.
- <String>: Record description.

Example

LIST:RECOrder 1,"0,1,100.0,110.0,0.0,60.0,1,0.0,180.0,1,1,1,20"

Query Syntax:

LIST:RECOrder? <NR1>

Returns:

<String>

LIST:RECOrder?

This command is used to query the record Arguments of a single Step in the List.

Syntax

LIST:RECOrder? <NR1>

Arguments:

<NR1>

<NR1>: Step index.

Returns:

<String>

<String>: The description of the Record is as follows:

<Step,Type,Wave Type,Level,Vac,Vdc,Freq,Time,Start Phase,Stop

Phase,Kac,Kdc,Kfreq,Repeat>

LIST:SAVE

This command is used to save the list file in a nonvolatile memory.

Syntax

LIST:SAVE <NR1>

Arguments:

1~5

Returns:

None

LIST:RECall

This command is used to recall a list file.

Syntax

LIST:RECall <NR1>

Arguments:

1~5

Returns:

None

LIST:RUN

This command is used to run the list file.

Syntax

LIST:RUN

Arguments:

None

Returns:

None

LIST:STOP

This command is used to stop the list file.

Syntax

LIST:STOP

Arguments:

None

Returns:

None

Chapter 10 SELFdefine Subsystem(IT-M7700)

SELFdefine commands is a special command of IT-M7721/IT-M7722/ IT-M7723/IT-M7723E/IT-M7722D/IT-M7722E/IT-M7723D, but it's not suitable for IT-M7721L/IT-M7722L/IT-M7723P.

One cycle of the selfdefine waveform is composed of 1024 Vi, and the value of each Vi ranges from -1 to 1. The AC voltage command of the power supply is based on the effective value of the selfdefine waveform, so when using the selfdefine waveform, the actual output voltage peak value will be adjusted according to the priority of the selfdefine waveform effective value. Calculated as follows:

Vrms(OUT) = |MAX(Vi)| * Vrms(SET)

Where:

- Vrms(OUT) is the actual output voltage value of the power supply.
- Vrms(SET) is the V-set value of the panel.
- |MAX(Vi)| is the maximum absolute value of -1 to 1 within i = 1 to 1024 points.

SELFdefine:NUMber

This command is used to set the current self-defined wave number.

Syntax

SELFdefine:NUMber

Arguments:

0~4

Query Syntax:

SELFdefine:NUMber?

Returns:

0~4

SELFdefine:NAME

This command is used to set the current self-defined wave name.

Syntax

SELFdefine:NAME

Arguments:

<String>

Query Syntax:

SELFdefine:NAME?

Returns:

<String>

SELFdefine:RECAll:NAME?

This command is used to query the saved self-defined wave file name.

Syntax

SELFdefine:RECAll:NAME? <NR1>

Arguments:

0~4

Returns:

<String>

SELFdefine:SAVE

This command is used to save the current self-defined wave file.

Syntax

SELFdefine:SAVE

Arguments:

None

SELFdefine:DATA

This command is used to set self-defined wave data.

Syntax

SELFdefine:DATA <NR1>,<NRf>

Arguments:

<NR1> Index（0-1023）

<NRf> Value（-1.0~1.0）

Query Syntax:

SELFdefine:DATA? <NR1>

Note: The store command (SELFdefine:SAVE) must have been executed before readback.

Returns:

<NRf>

SELFdefine:RECAll

This command is used to recall the self-defined file.

Syntax

SELFdefine:RECAll <String>

Arguments:

<String> => File name

Returns:

None

SELFdefine:EDIT

This command enables the user-defined waveforms.

Syntax

SELFdefine:EDIT

Arguments:

None

Returns:

None

Chapter 11 SELFdefine Subsystem(IT-M7723P)

The commands in this chapter are specific to the IT7723P and are not applicable to other models.

SELFdefine:USER:INDEX

This command is used to set the save address of the user-defined waveforms file.

Syntax

SELFdefine:USER:INDEX <Rn>

Arguments:

<Rn>0~4

Query Syntax:

SELFdefine:USER:INDEX?

Returns:

<Rn>

SELFdefine:USER:TYPE

This command is used to set the user-defined waveforms calculation type.

Syntax

SELFdefine:USER:TYPE

Arguments:

<0|1|2|OFF|THD|POINT>

Query Syntax:

SELFdefine:USER:TYPE?

Returns:

<OFF|THD|POINT>

- OFF: Clears the data at the location specified by SELFdefine:USER:INDEX.

Clearing is not completed until the SELFdefine:USER:POINT:SAVE or SELFdefine:USER:THD:SAVE command is executed.

- THD
- POINT

SELFdefine:USER:POINT:METHOD

This command is used to set the method of user-defined point waveforms.

Syntax

SELFdefine:USER:POINT:METHOD

Arguments:

<0|1|2|ASYMM|SYMM|POINTS>

Query Syntax:

SELFdefine:USER:POINT:METHOD?

Returns:

<ASYMM|SYMM|POINTS>

- ASYMM
- SYMM
- POINTS

SELFdefine:USER:POINT:LEN

This command is used to set the total number of points of user-defined point waveforms.

Syntax

SELFdefine:USER:POINT:LEN <Rn>

Arguments:

<Rn>1 ~ 1024

Query Syntax:

SELFdefine:USER:POINT:LEN?

Returns

<Rn>

SELFdefine:USER:POINT[:DATA]

This command is used to set the position data of user-defined point waveforms.

Syntax

SELFdefine:USER:POINT[:DATA]

Arguments:

<Rn>,<Level>

Rn: point position, 0 ~ 1023

Level: -1.0000 ~ 1.0000

Query Syntax:

SELFdefine:USER:POINT[:DATA]? <Rn>

Arguments: Rn: point position, 0 ~ 1023

Note: The store command (SELFdefine:USER:POINT:SAVE) must have been executed before readback.

Returns:

<Rn>,<Level>

Level: -1.0000 ~ 1.0000

Related commands:

SELFdefine:USER:POINT:SAVE

SELFdefine:USER:POINT:RECALL

SELFdefine:USER:POINT:SAVE

This command is used to store the current position information of the user-defined point waveforms.

Syntax

SELFdefine:USER:POINT:SAVE

Note: The storage delay time should be greater than 4000 milliseconds.

Arguments:

none

Query Syntax:

none

Related commands:

SELFdefine:USER:INDEX

SELFdefine:USER:POINT:RECALL

This command is used to read the specific position information of the user-defined point waveforms.

Syntax

SELFdefine:USER:POINT:RECALL <Rn>

Note: The readout delay time should be greater than (the total number of points in the user-defined waveform multiplied by 4 milliseconds).

Arguments:

<Rn>0 ~ 4

Query Syntax:

none

Related commands:

SELFdefine:USER:INDEX

SELFdefine:USER:POINT:LEN

SELFdefine:USER:THD:METHOD

This command is used to set the distortion factor calculation method of the user-defined THD waveforms.

Syntax

SELFdefine:USER:THD:METHOD

Arguments:

<0|1|THDF|THDR>

Query Syntax:

SELFdefine:USER:THD:METHOD?

Returns:

< THDF|THDR >

THDF: Total Harmonic Distortion (as % of fundamental).

THDR: Total Harmonic Distortion (as % of rms total).

THDF=sqrt(H2+H3+H4+...)/sqrt(H1）

THDR=sqrt(H2+H3+H4+...)/sqrt(H1+H2+H3+H4+...)

H1 represents the square of the fundamental frequency, H2 represents the square of the second harmonic, H3 represents the square of the third harmonic, and so on.

SELFdefine:USER:THD[:DATA]

This command is used to set the position data of the user-defined THD waveforms.

Syntax

SELFdefine:USER:THD[:DATA]

Arguments:

<Order>,<%>,<Phase>

Order: Thd order, 2~50

%: Amp, 0.00% ~ 100.00%

Phase: 0~ 359.9°

Query Syntax:

SELFdefine:USER:THD[:DATA]? <Order>

Arguments:Order: Thd order, 2~50

Note: The store command (SELFdefine:USER:THD:SAVE) must have been executed before readback.

Returns:

<%>,<Phase>

%: Amp, 0.00% ~ 100.00%

Phase: 0~ 359.9°

Related commands:

SELFdefine:USER:THD:SAVE

SELFdefine:USER:THD:RECALL

SELFdefine:USER:THD:SAVE

This command is used to store the current position information of the user-defined THD waveforms.

Syntax

SELFdefine:USER:THD:SAVE

Note: The storage delay time should be greater than 500 milliseconds.

Arguments:

none

Query Syntax:

none

Related commands:

SELFdefine:USER:INDEX

SELFdefine:USER:THD:RECALL

This command is used to read the specific position information of the user-defined THD waveforms.

Syntax

SELFdefine:USER:THD:RECALL <Rn>

Note: The readout delay time should be greater than 200 milliseconds.

Arguments:

<Rn>0 ~ 4

Query Syntax:

none

Related commands:

SELFdefine:USER:INDEX

Chapter 12 SWEEP command(IT-M7723P)

The commands in this chapter are specific to the IT7723P and are not applicable to other models.

Note: the Sweep waveform is selected with the following 3 commands.

- [SOURce:]NORMal:WAVE
- [SOURce:]NORMal:WAVE:THD
- [SOURce:]NORMal:WAVE:USER

SWEep:VOLTage:STARt:AC

This command is used to set the start AC voltage of Sweep.

Syntax

SWEep:VOLTage:STARt:AC <NRf>

Arguments:

<NRf>, 0 ~ 305

Query Syntax:

SWEep:VOLTage:STARt:AC?

Returns:

<NRf>

SWEep:VOLTage:STOP:AC

This command is used to set the stop AC voltage of Sweep.

Syntax

SWEep:VOLTage:STOP:AC <NRf>

Arguments:

<NRf>

0 ~ 305

Query Syntax:

SWEep:VOLTage:STOP:AC?

Returns:

<NRf>

SWEep:VOLTage:STEP:AC

This command is used to set the step AC voltage of Sweep.

Syntax

SWEep:VOLTage:STEP:AC <NRf>

Arguments:

<NRf>

0 ~ 305

Query Syntax:

SWEep:VOLTage:STEP:AC?

Returns:

<NRf>

SWEep:VOLTage:STARt:DC

This command is used to set the start DC voltage of Sweep.

Syntax

SWEep:VOLTage:STARt:DC <NRf>

Arguments:

<NRf>

Range: NORMal:VOLTage:DC:MIN ~ NORMal:VOLTage:DC:MAX

Query Syntax:

SWEep:VOLTage:STARt:DC?

Returns:

<NRf>

SWEep:VOLTage:STOP:DC

This command is used to set the stop DC voltage of Sweep.

Syntax

SWEep:VOLTage:STOP:DC <NRf>

Arguments:

<NRf>

Range: NORMal:VOLTage:DC:MIN ~ NORMal:VOLTage:DC:MAX

Query Syntax:

SWEep:VOLTage:STOP:DC?

Returns:

<NRf>

SWEep:VOLTage:STEP:DC

This command is used to set the step DC voltage of Sweep.

Syntax

SWEep:VOLTage:STEP:DC <NRf>

Arguments:

<NRf>

Range: NORMal:VOLTage:DC:MIN ~ NORMal:VOLTage:DC:MAX

Query Syntax:

SWEep:VOLTage:STEP:DC?

Returns:

<NRf>

SWEep:FREQ:STARt

This command is used to set the start frequency of Sweep.

Syntax

SWEep:FREQ:STARt <NRf>

Arguments:

<NRf>

45 ~ 1000

Query Syntax:

SWEep:FREQ:STARt?

Returns:

<NRf>

SWEep:FREQ:STOP

This command is used to set the stop frequency of Sweep.

Syntax

SWEep:FREQ:STOP <NRf>

Arguments:

<NRf>

45 ~ 1000

Query Syntax:

SWEep:FREQ:STOP?

Returns:

<NRf>

SWEep:FREQ:STEP

This command is used to set the step frequency of Sweep.

Syntax

SWEep:FREQ:STEP <NRf>

Arguments:

<NRf>

Range: 0.1 ~ NORMal:FREQuency:MAX

Query Syntax:

SWEep:FREQ:STEP?

Returns:

<NRf>

SWEep:TIMe:STEP

This command is used to set the step time of Sweep.

Syntax

SWEep:TIMe:STEP <NRf>

Arguments:

<NRf>

Unit: s.

Query Syntax:

SWEep:TIMe:STEP?

Returns:

<NRf>

SWEep:MODe

This command is used to set the drop mode of Sweep.

Syntax

SWEep:MODe <mode>

Arguments:

TIME|TRIG|TIME_LOOP|TRIG_LOOP

TIMe_LOOP is a time loop scan.

TRIG_LOOP is a trigger loop scan.

Query Syntax:

SWEep:MODe?

Returns:

< TIME|TRIG|TIME_LOOP|TRIG_LOOP >

SWEep:RUN

This command is used to start Sweep.

Syntax

SWEep:RUN

Arguments:

None

Query Syntax:

None

SWEep:STOP

This command is used to stop Sweep.

Syntax

SWEep:STOP

Arguments:

None

Query Syntax:

None

SWEep:PRIority

This command is used to set the priority of Sweep.

Syntax

SWEep:PRIority < priority>

Arguments:

VOLT|FREQ|VF

Query Syntax:

SWEep:PRIority?

Returns:

<VOLT|FREQ|VF >

SWEep:FINish

This command is used to set the end state of Sweep.

Syntax

SWEep:FINish < mode>

Arguments:

OFF|LAST|NORM

Query Syntax:

SWEep:FINish?

Returns:

< OFF|LAST|NORM >

SWEep:STEP:REPeat

This command is used to set the number of repetitions of sweep, set 0 for infinite loop.

Syntax

SWEep:STEP:REPeat <RN1>

Arguments:

< RN1>

0: infinite loop.

1 ~ 999999

Query Syntax:

SWEep: STEP:REPeat?

Returns:

< RN1>

0: infinite loop.

1 ~ 999999

SWEep:STATe?

This command is used to query the running status of Sweep.

Syntax

SWEep:STATe?

Arguments:

None

Returns:

0:Stop, 1:Waiting for trigger, 2:Running.

Chapter 13 STANdard command(IT-M7723P)

The commands in this chapter are specific to the IT7723P and are not applicable to other models.

STANdard:CATegory:TYPE

This command is used to select the category of regulations.

Syntax

STANdard:CATegory:TYPE <Category>,<Class>

Arguments:

<Category>	<Class>
0: Voltage dips	0:Class2 1:Class3 2:User define
1: short interruptions	0:Class2 1:Class3 2:User define
2: Voltage variations	0:70% 1: User define
3: Flat curve	0:Class1 1:Class2 2:Class3 3:User define
4: Over swing	0:Class1 1:Class2 2:Class3 3:User define
5: Sweep in frequency	0:Class1 1:Class2 2:Class3 3:User define
6: Odd non-multiple of 3 harmonics	0:Class1 1:Class2 2:Class3 3:User define
7: Odd multiple of 3 harmonics	0:Class1 1:Class2 2:Class3 3:User define
8: Even harmonics	0:Class1 1:Class2 2:Class3 3:User define
9: Interharmonics	0:Class2 1:Class3 2:User define
10: Meister curve	0:Class2 1:Class3 2:User define
11 :Voltage fulctuations	0:Class2 1:Class3 2:User define
12 :Frequency variations	0:Class2 1:Class3 2:Class4 3:User define

Query Syntax:

STANdard:CATegory:TYPE?

Returns:

<Category>,<Class>

STANdard:RECALL

This command is used to read back the regulation category data

(synchronization). A delay of 2 seconds is required after executing this command.

Syntax

STANdard:RECALL

Arguments:

None

Returns:

None

STANdard:CATegory:VOLTage:FREQuency

This command is used to set the voltage and frequency for regulatory testing.

Syntax

STANdard:CATegory:VOLTage:FREQuency <RNf1>,<RNf2>

Arguments:

<RNf1> test voltage

<RNf2> test frequency

Query Syntax:

STANdard:CATegory:VOLTage:FREQuency?

Returns:

< RNf1>,<RNf2>

STANdard:CATegory:VOLTage:TR

This command is used to set the voltage rise time for regulatory testing (Voltage dips).

Syntax

STANdard:CATegory:VOLTage:TR <RN1>,<RNf>

Arguments:

<RN1> 0 ~ 2

0: Class2, 1: Class3, 2: User define

< RNf> 0.2 ~ 1000.0

Query Syntax:

STANdard:CATegory:VOLTage:TR? <RN1>

Arguments:

<RN1> 0 ~ 2

0: Class2, 1: Class3, 2: User define

Returns:

<RNf>

STANdard:CATegory:VOLTage:TF

This command is used to set the voltage fall time for regulatory testing (Voltage dips).

Syntax

STANdard:CATegory:VOLTage:TF <RN1>,<RNf>

Arguments:

<RN1> 0 ~ 2

0: Class2, 1: Class3, 2: User define

< RNf> 0.2 ~ 1000.0

Query Syntax:

STANdard:CATegory:VOLTage:TF? <RN1>

Arguments:

<RN1> 0 ~ 2

0: Class2, 1: Class3, 2: User define

Returns:

< RNf>

STANdard:CATegory:PERiod

This command is used to set the test period for regulatory testing (Voltage fulctuations).

Syntax

STANdard:CATegory:PERiod <RN1>,<RNf>

Arguments:

<RN1> 0 ~ 2

0: Class2, 1: Class3, 2: User define

< RNf> 0 ~ 10000

Query Syntax:

STANdard:CATegory:PERiod? <RN1>

Arguments:

<RN1> 0 ~ 2

0: Class2, 1: Class3, 2: User define

Returns:

< RNf>

STANdard:CATegory:DURation

This command is used to set the duration of regulatory testing (Voltage fulctuations).

Syntax

STANdard:CATegory:DURation <RN1>,<RNf>

Arguments:

<RN1> 0 ~ 2

0: Class2, 1: Class3, 2: User define

< RNf>0.2 ~ 10000

Query Syntax:

STANdard:CATegory: DURation? <RN1>

Arguments:

<RN1> 0 ~ 2

0: Class2, 1: Class3, 2: User define

Returns:

< RNf>

STANdard:CATegory:RUNTime

This command is used to set the run time for regulatory testing (Over swing).

Syntax

STANdard:CATegory:RUNTime <RN1>,<RNf>

Arguments:

<RN1> 0 ~ 3

0: Class1, 1: Class2, 2: Class3, 3: User define

< RNf> 0 ~ 120000

Query Syntax:

STANdard:CATegory: RUNTime? <RN1>

Arguments:

<RN1> 0 ~ 3

0: Class1, 1: Class2, 2: Class3, 3: User define

Returns:

< RNf>

STANdard:VOLT:DIP

This command is used to set the parameters of IEC 61000-4-11 Voltage dips test.

Syntax

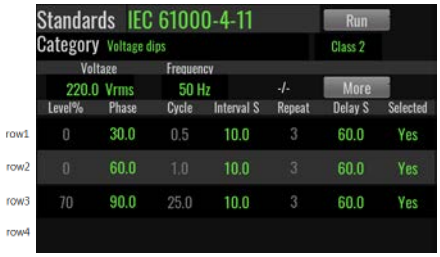
STANdard:VOLT:DIP <Class><line>,<Select>,<level>,<phase>,<cycle>,<IntervalS>,<Repeat>,<DelayS>

Arguments:

Class 2/Class 3	<class>,<row>,<Select>,<phase>,<IntervalS>,<DelayS>
User define	<Class>,<row>,<Select>,<level>,<phase>,<cycle>,<IntervalS>,<Repeat>,<DelayS>

< Class> 0: Class2, 1: Class3, 2: User define

<row> 0: row1, 1: row2, 2: row3, 3: row4, 4: row5



<select> 0:No 1:Yes

<level> 0~100

<phase> 0.0~360.0

<cycle> 0.0~2000.0

<IntervalS> 10~2000

<Repeat> 1~1000

<DelayS> 0.0~60.0

Query Syntax:

STANdard:VOLT:DIP? <class>,<row>

Returns:

<row>,<Select>,<level>,<phase>,<cycle>,<IntervalS>,<Repeat>,<DelayS>

STANdard:SHORT:INTER

This command is used to set the parameters of IEC 61000-4-11 Voltage short interruptions test.

Syntax

STANdard:SHORT:INTER <Class>,<level>,<phase>,<cycle>,<IntervalS>,<Repeat>

Arguments:

Class 2/Class 3	<class>,<phase>,<IntervalS>
User define	<Class>,<level>,<phase>,<cycle>,<IntervalS>,<Repeat>

< Class> 0: Class2, 1: Class3, 2: User define

<level> 0.0 ~ 100.0

<phase> 0.0~360.0

<cycle> 0.0~2000.0

<IntervalS> 10.0~2000.0

<Repeat> 1~1000

Query Syntax:

STANdard:SHORT:INTER? <Class>

Returns:

<level>,<phase>,<cycle>,<IntervalS>,<Repeat>

STANdard:VOLT:VAR

This command is used to set the parameters of IEC 61000-4-11 Voltage variations test.

Syntax

STANdard:VOLT:VAR <Class>,<level>,<phase>,<Td>,<Ts>,<Ti>,<Repeat>,<IntervalS>

Arguments:

70%	<class>,<phase>
User define	<Class>,<level>,<phase>,<Td>,<Ts>,<Ti>,<Repeat>,<IntervalS>

< Class> 0: 70%, 1: User define

<level> 0 ~ 99

<phase> 0.0~360.0

<Td> 0~500

<Ts> 0~500

<Ti> 0~500

<Repeat> 1~1000

<IntervalS> 10.0~2000.0

Query Syntax:

STANdard:VOLT:VAR? <Class>

Returns:

<level>,<phase>,<Td>,<Ts>,<Ti>,<Repeat>,<IntervalS>

STANdard:FLAT:CUR

This command is used to set the parameters of IEC 61000-4-13 Flat curve test.

Syntax

STANdard:FLAT:CUR <Class>,<Flat_curve>,<Mode>,<Time>

Arguments:

Class1/Class2/Class3	<Class>
User define	<Class>,<Flat_curve>,<Mode>,<Time>

<Class> 0: Class1, 1: Class2, 2: Class3, 3: User define

<Flat_curve> 5 ~ 95

<Mode> 0: Infinity, 1: Time

<Time> 0.100 ~ 500.000

Query Syntax:

STANdard:FLAT:CUR? <Class>

Returns:

<Flat_curve>,<Volt_ratio_Ky>,<Mode>,<Time>

STANdard:OVER:SW

This command is used to set the parameters for the IEC 61000-4-13 Over swing test.

Syntax

STANdard:OVER:SW <Class>,<3rdthd>,<3rdphase>,<5ththd>,<5thphase>

Arguments:

Class1/Class2/Class3	<Class>
User define	<Class>,<3rdthd>,<3rdphase>,<5ththd>,<5thphase>

< Class> 0: class 1, 1: class 2, 2: class 3, 3: User define

<3rdthd> 0.0~20.0

<3rdphase> 0.0~360.0

<5ththd> 0.0~20.0

<5thphase> 0.0~360.0

Query Syntax:

STANdard:OVER:SW? <Class>

Returns:

<3rdthd>,<3rdphase>,<5ththd>,<5thphase>

STANdard:SW:FREQ

This command sets the parameters for the IEC 61000-4-13 Sweep in frequency test.

Syntax

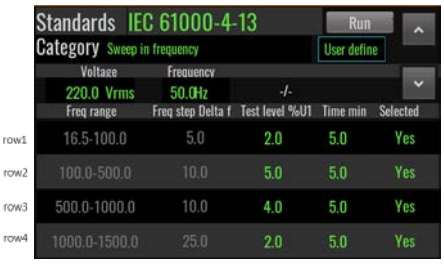
STANdard:SW:FREQ <Class>,<row>,<select>,<Test level>,<Time min>

Arguments:

Class1/Class2/Class3	<Class>,<row>,<select>
User define	<Class>,<row>,<select>,<Test level>,<Time min>

< Class> 0: Class1, 1: Class2, 2: Class3, 3: User define

<row> 0: row1, 1: row2, 2: row3, 3: row4, 4: row5



<select> 0: No, 1: Yes

<Test level> 0.0~100.0

<Time min> 0.0~100.0

Query Syntax:

STANdard:SW:FREQ? <Class>,<row>

Returns:

<row>,<select>,<Test level>,<Time min>

STANdard:INDI:HARM:ODD:NON3

This command is used to set the test parameters for IEC 61000-4-13 Odd non-multiple of 3 Harmonic.

Syntax

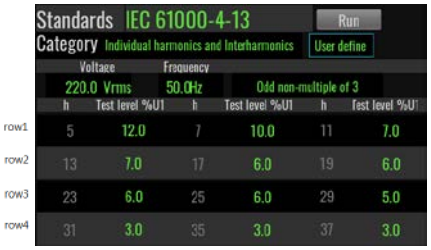
STANdard:INDI:HARM:ODD:NON3 <Class>,<row>,<Test levelU 1>,<Test levelU 2>,<Test levelU 3>

Arguments:

Class1/Class2/Class3	<Class>
User define	<Class>,<row>,<Test levelU 1>,<Test levelU 2>,<Test levelU 3>

< Class> 0: Class1, 1: Class2, 2: Class3, 3: User define

<row> 0: row1, 1: row2, 2: row3, 3: row4



<Test levelU 1> 0.0 ~ 100.0

<Test levelU 2> 0.0 ~ 100.0

<Test levelU 3> 0.0 ~ 100.0

Query Syntax:

STANdard:INDI:HARM:ODD:NON3? <Class>,<row>

Returns:

<row>,<h1>,<Test levelU 1>,<h2>,<Test levelU 2>,<h3>,<Test levelU 3>

STANdard:INDI:HARM:ODD3

This command is used to set the test parameters for IEC 61000-4-13 Odd multiple of 3 Harmonic.

Syntax

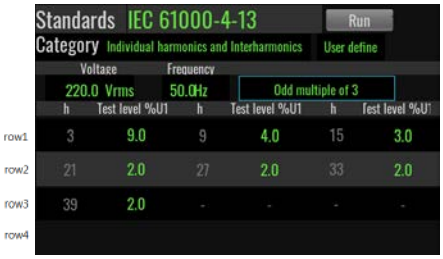
STANdard:INDI:HARM:ODD3 <Class>,<row>,<Test levelU 1>,<Test levelU 2>,<Test levelU 3>

Arguments:

Class1/Class2/Class3	<Class>
User define	<Class>,<row>,<Test levelU 1>,<Test levelU 2>,<Test levelU 3>

< Class> 0: Class1, 1: Class2, 2: Class3, 3: User define

<row> 0: row1, 1: row2, 2: row3



<Test levelU 1> 0.0 ~ 100.0

<Test levelU 2> 0.0 ~ 100.0

<Test levelU 3> 0.0 ~ 100.0

Query Syntax:

STANdard:INDI:HARM:ODD3? <Class>,<row>

Returns:

<row>,<h1>,<Test levelU 1>,<h2>,<Test levelU 2>,<h3>,<Test levelU 3>

STANdard:INDI:HARM:EVEN

This command is used to set the test parameters for IEC 61000-4-13 Even Harmonic.

Syntax

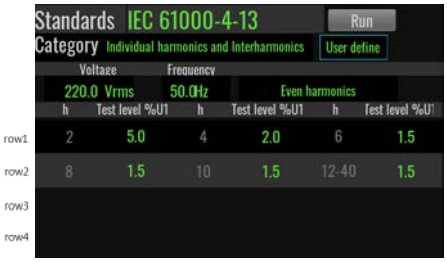
STANdard:INDI:HARM:EVEN <Class>,<row>,<Test levelU 1>,<Test levelU 2>,<Test levelU 3>

Arguments:

Class1/Class2/Class3	<Class>
User define	<Class>,<row>,<Test levelU 1>,<Test levelU 2>,<Test levelU 3>

< Class> 0: Class1, 1: Class2, 2: Class3, 3: User define

<row> 0: row1, 1: row2



<Test levelU 1> 0.0 ~ 100.0

<Test levelU 2> 0.0 ~ 100.0

<Test levelU 3> 0.0 ~ 100.0

Query Syntax:

STANdard:INDI:HARM:EVEN? <Class>,<row>

Returns:

<row>,<h1>,<Test levelU 1>,<h2>,<Test levelU 2>,<h3>,<Test levelU 3>

STANdard:INTER:HARM

This command is used to set the test parameters for IEC 61000-4-13 Even Harmonic.

Syntax

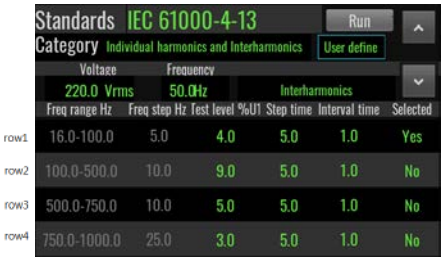
STANdard:INTER:HARM <Class>,<row>,<Select>,<Test levelU 1>,<Step time>,<Interval time>

Arguments:

Class1/Class2/Class3	<Class>,<row>,<Select>
User define	<Class>,<row>,<Select><Test levelU 1>,<Step time>,<Interval time>

<Class> 0: Class2, 1: Class3, 2: User define

<row> 0: row1, 1: row2, 2: row3, 3: row4, 4: row5



<Select> 0: No, 1: Yes

<Test levelU 1> 0.0 ~ 100.0

<Step time> 0.0 ~ 1000.0

<Interval time> 0.0 ~ 1000.0

Query Syntax:

STANdard:INTER:HARM? <Class>,<row>

Returns:

<row>,<Select>,<Freq range>,<Freq step>,<Test levelU 1>,<Step time>,<Interval time>

STANdard:MEI:CUR

This command is used to set the test parameters for IEC 61000-4-13 Meister curve.

Syntax

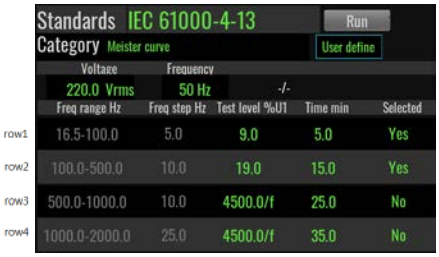
STANdard:MEI:CUR <Class>,<row>,<Select>,<Test levelU 1>,<Step time>,<Interval time>

Arguments:

Class1/Class2/Class3	<Class>,<row>,<Select>
User define	<Class>,<row>,<Select><Test levelU 1>,<Time min>

< Class> 0: Class2, 1: Class3, 2: User define

< row> 0: row1, 1: row2, 2: row3, 3: row4



<Select> 0:No 1:Yes

<Test levelU 1>

row1,row2 0.0 ~ 100.0

row3,row4 0.0 ~ 10000.0

<Time min> 0.0 ~ 100.0

Query Syntax:

STANdard:MEI:CUR? <Class>,<row>

Returns:

<row>,<Select>,<Freq range>,<Freq step>,<Test levelU 1>,<Time min>

STANdard:VOLT:FULCT

This command is used to set the test parameters for IEC 61000-4-14 Voltage fluctuations.

Syntax

STANdard:VOLT:FULCT <Class>,<row>,<Select>,<Level>,<△U>,<Repeat>,<DelayS>

Arguments:

Class1/Class2/Class3	<Class>,<row>,<Select>,
User define	<Class>,<row>,<Select>,<Level>,<△U>,<Repeat>,<DelayS>

< Class> 0: Class2, 1: Class3, 2: User define

<row> 0: row1, 1: row2, 2: row3

<Select> 0:No 1:Yes

<Level> 0 ~ 200

<△U> If the Level is less than 100, the range is from 0 to 20.

If the Level is greater than 100, the range is from -20 to 0.

If the Level equals 100, the range is either -20 to 0 or 0 to 20.

<Repeat> 1 ~ 100

<DelayS> 60.000 ~ 999.900

Query Syntax:

STANdard:VOLT:FULCT? <Class>,<row>

Returns:

<row>,<Select>,<Level>,<△U>,<Repeat>,<DelayS>

STANdard:FREQ:VAR

This command is used to set the test parameters for IEC 61000-4-28 Frequency variations.

Syntax

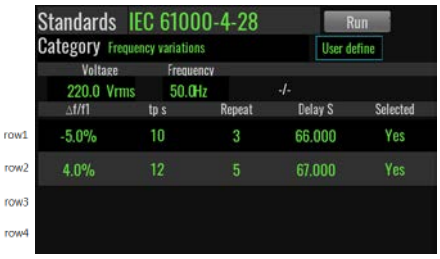
STANdard:FREQ:VAR <Class>,<row>,<Select>,<Level>,<△U>,<Repeat>,<DelayS>

Arguments:

Class1/Class2/Class3	<Class>,<row>,<Select>,<DelayS>
User define	<Class>,<row>,<Select>,<△f/f1>,<tps>,<Repeat>,<DelayS>

< Class> 0: Class2, 1: Class3, 2: Class4, 3: User define

<row> 0: row1, 1: row2



<Select> 0: No, 1: Yes

<△f/f1>

f1: Test frequency, test frequency range from fL (40Hz) to fH (70Hz).

△f: (Lowest frequency fL - Test frequency f1) ~ (Highest frequency fH - Test frequency f1).

The range of △f/f1 is from (fL - f1)/f1 % to (fH - f1)/f1 %.

<tps> User define, 1 ~ 360

<Repeat> User define, 1 ~ 100

<DelayS> 60.000 ~ 999.900

Query Syntax:

STANdard:FREQ:VAR? <Class>,<row>

Returns:

<row>,<Select>,<△f/f1>,<tps>,<Repeat>,<DelayS>

STANdard:RUN

The command is used to run the selected regulatory category.

Syntax

STANdard:RUN

Arguments:

None

Returns:

None

STANdard:STOP

The command is used to stop the running selected regulatory category.

Syntax

STANdard:RUN

Arguments:

None

Returns:

None

STANdard:STATe?

The command is used to query the running status of regulatory categories.

Syntax

STANdard:STATe?

Arguments:

None

Returns:

<current row>,<current repeat>,<Stop Run>

<current row>: Which row is currently running. 0: row1, 1: row2, 2: row3, 3: row4, 4: row5.

<current repeat>: The repeat times of current row.

<Stop Run> 0: Stop, 1: Run

Chapter 14 Multi-Channel command(IT-M7723)

The commands in this chapter are specific to the IT7723 and are not applicable to other models.

CHANnel

This command is used to set the address of the host computer to send data to the online instrument.

Syntax

CHANnel <NR1>

Arguments

<NR1> 1 ~ 126 (1 to 1 transmission)

127 (1 to multi transmission)

Query Syntax

CHANnel?

Returns

<NR1>

Note

For the multi-channel operation method, please refer to the content of the multi-channel function chapter of "IT-M7700 User Manual".

CHANnel:LINK

This command is used to determine whether there is a conflict between the IDs of the master and the slaves.

Syntax

CHANnel:LINK

Arguments

None

Note

The detection time of this command takes 1500ms, so the execution of this command requires a delay of 1.5 seconds. ID refers to the menu Config->Channel Number.

Before this command is executed, the CHAN 127 command must be executed first.

Also see

SYST:REM *OPC *OPC? CHAN:ERR?

CHANnel:ERRor?

This command is used to read the state after CHANnel:LINK is executed.

Syntax

CHANnel:ERRor?

Returns

0: No ID conflict, multi-channel function can be operated normally

1: Slave Master ID Conflict

2: Slave Slave ID Conflict

Note

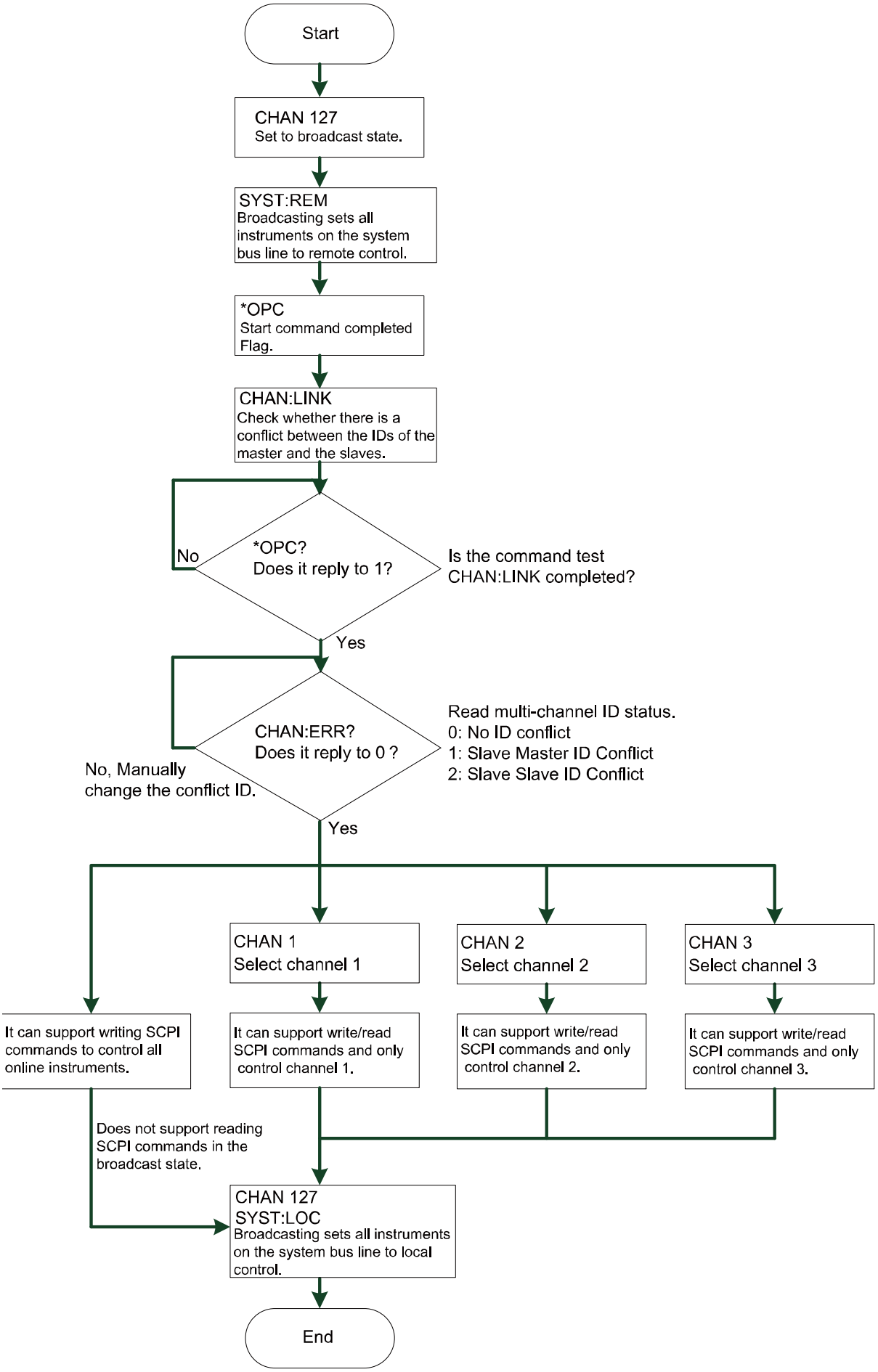
ID refers to the menu Config->Channel Number. If there is a conflict, please press the ESC key on the display panel to cancel and manually change the ID.

Also see

CHANnel:LINK

Multi-channel program flow

The operation flowchart of multi-channel program control is as follows.


Start
CHAN 127
Set to broadcast state.
SYST:REM
Broadcasting sets all instruments on the system bus line to remote control.
*OPC
Start command completed Flag.
CHAN:LINK
Check whether there is a conflict between the IDs of the master and the slaves.
No
*OPC?
Does it reply to 1?
Is the command test CHAN:LINK completed?
Yes
CHAN:ERR?
Does it reply to 0 ?
Read multi-channel ID status.
0: No ID conflict
1: Slave Master ID Conflict
2: Slave Slave ID Conflict
No, Manually change the conflict ID.
Yes
CHAN 1
Select channel 1
CHAN 2
Select channel 2
CHAN 3
Select channel 3
It can support writing SCPI commands to control all online instruments.
It can support write/read SCPI commands and only control channel 1.
It can support write/read SCPI commands and only control channel 2.
It can support write/read SCPI commands and only control channel 3.
Does not support reading SCPI commands in the broadcast state.
CHAN 127
SYST:LOC
Broadcasting sets all instruments on the system bus line to local control.
End

The following commands are suitable for IT-M7723 multi-channel applications (support write/read SCPI commands).

Since it is a multi-channel application, the interval between each command is at least 30 mS. *IDN? at least 60mS.

If the interface card is RS232 (IT-E1207), it may be necessary to extend the command interval due to different baud rates.

*IDN?

SYSTem:REMote

SYSTem:LOCal

SYSTem:RWLock

SYSTem:BEEPer

SYSTem:BEEPer?

SYSTem:POWnon

SYSTem:POWnon?

[SOURce:]RELAy:MODe

[SOURce:]RELAy:MODe?

[SOURce:]NORMal:VOLTage:AC[:LEVel][:IMMediate][:AMPLitude]

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

[SOURce:]NORMal:VOLTage:DC[:LEVel][:IMMediate]

[SOURce:]NORMal:VOLTage:DC[:LEVel][:IMMediate]?

[SOURce:]NORMal:FREQuency[:LEVel][:IMMediate]

[SOURce:]NORMal:FREQuency[:LEVel][:IMMediate]?

[SOURce:]NORMal:VOLTage:AC:MAX[:LEVel]

[SOURce:]NORMal:VOLTage:AC:MAX[:LEVel]?

[SOURce:]NORMal:VOLTage:AC:MIN[:LEVel]

[SOURce:]NORMal:VOLTage:AC:MIN[:LEVel]?

[SOURce:]NORMal:VOLTage:DC:MAX[:LEVel]

[SOURce:]NORMal:VOLTage:DC:MAX[:LEVel]?

[SOURce:]NORMal:VOLTage:DC:MIN[:LEVel]

[SOURce:]NORMal:VOLTage:DC:MIN[:LEVel]?

[SOURce:]NORMal:FREQuency:MAX[:LEVel]

[SOURce:]NORMal:FREQuency:MAX[:LEVel]?

[SOURce:]NORMal:FREQuency:MIN[:LEVel]

[SOURce:]NORMal:FREQuency:MIN[:LEVel]?

[SOURce:]NORMal:PHASe:STARt[:LEVel][:IMMediate]

[SOURce:]NORMal:PHASe:STARt[:LEVel][:IMMediate]?

[SOURce:]NORMal:PHASe:STOP[:LEVel][:IMMediate]

[SOURce:]NORMal:PHASe:STOP[:LEVel][:IMMediate]?

[SOURce:]NORMal:MODE

[SOURce:]NORMal:MODE?

[SOURce:]OUTPut[:STATe]

[SOURce:]OUTPut[:STATe]?

[SOURce:]NORMal:VRISETIME

[SOURce:]NORMal:VRISETIME?

[SOURce:]NORMal:VOLTage:RANGe

[SOURce:]NORMal:VOLTage:RANGe?

[SOURce:]NORMal:WAVE

[SOURce:]NORMal:WAVE?

[SOURce:]NORMal:WAVE:CSINe

[SOURce:]NORMal:WAVE:CSINe?

[SOURce:]NORMal:DIMMer:MODe

[SOURce:]NORMal:DIMMer:MODe?

[SOURce:]NORMal:DIMMer[:PHASe]

[SOURce:]NORMal:DIMMer[:PHASe]?

[SOURce:]NORMal:SURGETRAP:MODe

[SOURce:]NORMal:SURGETRAP:MODe?

[SOURce:]NORMal:SURGETRAP:PERIOD

[SOURce:]NORMal:SURGETRAP:PERIOD?

[SOURce:]NORMal:SURGETRAP:WIDTH

[SOURce:]NORMal:SURGETRAP:WIDTH?

[SOURce:]NORMal:SURGETRAP:PERCENT

[SOURce:]NORMal:SURGETRAP:PERCENT?

PROTect:RMS:VOLTage

PROTect:RMS:VOLTage?

PROTect:PEAK:VOLTage

PROTect:PEAK:VOLTage?

PROTect:RMS:UNVOLTage

PROTect:RMS:UNVOLTage?

PROTect:RMS:CURRent

PROTect:RMS:CURRent?

PROTect:PEAK:CURRent

PROTect:PEAK:CURRent?

PROTect:RMS:CURRent:TIME

PROTect:RMS:CURRent:TIME?

PROTect:RMS:CURRent:MAX[:LEVel]

PROTect:RMS:CURRent:MAX[:LEVel]?

PROTect:RMS:CURRent:MIN[:LEVel]

PROTect:RMS:CURRent:MIN[:LEVel]?

PROTect:POWer

PROTect:POWer?

PROTect:MAX:CURRent:LIMit

PROTect:MAX:CURRent:LIMit?

PROTect:CLEar

LIST:TRIGger:MODe

LIST:TRIGger:MODe?

LIST:STATe

LIST:STATe?

LIST:RUN

LIST:STOP

FETCh[:SCALar]:VOLTage:AC?

FETCh[:SCALar]:VOLTage:DC?

FETCh[:SCALar]:CURRent:AC?

FETCh[:SCALar]:CURRent:DC?

FETCh[:SCALar]:POWer[:REAL]?

FETCh[:SCALar]:POWer:APParent?

FETCh[:SCALar]:POWer:PFACtor?

FETCh[:SCALar]:FREQuency?

FETCh[:SCALar]:CURRent:PEAK?

FETCh[:SCALar]:THD?

FETCh[:SCALar]:POWer:REACTive?

MEASure[:SCALar]:VOLTage:AC?

MEASure [:SCALar]:VOLTage:DC?

MEASure [:SCALar]:CURRent:AC?

MEASure [:SCALar]:CURRent:DC?

MEASure [:SCALar]:POWer[:REAL]?

MEASure [:SCALar]:POWer:APParent?

MEASure [:SCALar]:POWer:PFACtor?

MEASure [:SCALar]:FREQuency?

MEASure [:SCALar]:CURRent:PEAK?

MEASure [:SCALar]:THD?

MEASure [:SCALar]:POWer:REACTive?

Chapter 15 FETCh & MEASure Subsystem

FETCh[:SCALar]:VOLTage:AC?

This command reads the Vrms value.

Syntax

FETCh[:SCALar]:VOLTage:AC?

Arguments:

None

Returns:

< NRf >

FETCh[:SCALar]:VOLTage:DC?

This command reads the Vdc value.

Syntax

FETCh[:SCALar]:VOLTage:DC?

Arguments:

None

Returns:

< NRf >

FETCh[:SCALar]:CURRent:AC?

This command reads the Irms value.

Syntax

FETCh[:SCALar]:CURRent:AC?

Arguments:

None

Returns:

< NRf >

FETCh[:SCALar]:CURRent:DC?

This command reads the Idc value.

Syntax

FETCh[:SCALar]:CURRent:DC?

Arguments:

None

Returns:

< NRf >

FETCh[:SCALar]:POWer[:REAL]?

This command reads the active power.

Syntax

FETCh[:SCALar]:POWer[:REAL]?

Arguments:

None

Returns:

< NRf >

FETCh[:SCALar]:POWer:APParent?

This command reads the apparent power.

Syntax

FETCh[:SCALar]:POWer:APParent?

Arguments:

None

Returns:

< NRf >

FETCh[:SCALar]:POWer:PFACtor?

This command reads the power factor.

Syntax

FETCh[:SCALar]:POWer:PFACtor?

Arguments:

None

Returns:

< NRf >

FETCh[:SCALar]:FREQuency?

This command reads the frequency.

Syntax

FETCh[:SCALar]:FREQuency?

Arguments:

None

Returns:

< NRf >

FETCh[:SCALar]:CURRent:PEAK?

This command reads the peak current.

Syntax

FETCh[:SCALar]:CURRent:PEAK?

Arguments:

None

Returns:

< NRf >

FETCh[:SCALar]:THD?

This command reads the voltage harmonic distortion.

Syntax

FETCh[:SCALar]:THD?

Arguments:

None

Returns:

< NRf >

FETCh[:SCALar]:CURRent:THD?

This command reads the current harmonic distortion.

Syntax

FETCh[:SCALar]:CURRent:THD?

Arguments:

None

Returns:

< NRf >

FETCh[:SCALar]:POWer:REACTive?

This command reads the reactive power.

Syntax

FETCh[:SCALar]:POWer:REACTive?

Arguments:

None

Returns:

< NRf >

MEASure[:SCALar]:VOLTage:AC?

This command reads the Vrms value.

Syntax

MEASure[:SCALar]:VOLTage:AC?

Arguments:

None

Returns:

< NRf >

MEASure[:SCALar]:VOLTage:DC?

This command reads the Vdc value.

Syntax

MEASure[:SCALar]:VOLTage:DC?

Arguments:

None

Returns:

< NRf >

MEASure[:SCALar]:CURRent:AC?

This command reads the Irms value.

Syntax

MEASure[:SCALar]:CURRent:AC?

Arguments:

None

Returns:

< NRf >

MEASure[:SCALar]:CURRent:DC?

This command reads the Idc value.

Syntax

MEASure[:SCALar]:CURRent:DC?

Arguments:

None

Returns:

< NRf >

MEASure[:SCALar]:POWer[:REAL]?

This command reads the active power.

Syntax

MEASure[:SCALar]:POWer[:REAL]?

Arguments:

None

Returns:

< NRf >

MEASure[:SCALar]:POWer:APParent?

This command reads the apparent power.

Syntax

MEASure[:SCALar]:POWer:APParent?

Arguments:

None

Returns:

< NRf >

MEASure[:SCALar]:POWer:PFACtor?

This command reads the power factor.

Syntax

MEASure[:SCALar]:POWer:PFACtor?

Arguments:

None

Returns:

< NRf >

MEASure[:SCALar]:FREQuency?

This command reads the frequency.

Syntax

MEASure[:SCALar]:FREQuency?

Arguments:

None

Returns:

< NRf >

MEASure[:SCALar]:THD?

This command reads the voltage harmonic distortion.

Syntax

MEASure[:SCALar]:THD?

Arguments:

None

Returns:

< NRf >

MEASure[:SCALar]:CURRent:THD?

This command reads the current harmonic distortion.

Syntax

MEASure[:SCALar]:CURRent:THD?

Arguments:

None

Returns:

< NRf >

MEASure[:SCALar]:POWer:REACTive?

This command reads the reactive power.

Syntax

MEASure[:SCALar]:POWer:REACTive?

Arguments:

None

Returns:

< NRf >

MEASure?

This command is used to re-read the values of Arguments such as Vrms, Vdc, Irms, Idc, Ipk_plus, Ipk_minus, W (Real Power), PF, IPmax, VA, Q(VAR), VTHD, Freq, Vpeak, VAC, IAC, ITHD.

Syntax

MEASure?

Arguments:

None

Returns:

< Vrms, Vdc, Irms, Idc, Ipk_plus, Ipk_minus, W(Real Power), PF, IPmax, VA, Q(VAR), VTHD, Freq, Vpeak, VAC, IAC, ITHD>

FETCh?

This command is used to read the values of Arguments such as Vrms, Vdc, Irms, Idc, Ipk_plus, Ipk_minus, W (Real Power), PF, IPmax, VA, Q(VAR), VTHD, Freq, Vpeak, VAC, IAC, ITHD.

Syntax

FETCh?

Arguments:

None

Returns:

< Vrms, Vdc, Irms, Idc, Ipk_plus, Ipk_minus, W(Real Power), PF, IPmax, VA, Q(VAR), VTHD, Freq, Vpeak, VAC, IAC, ITHD >

Note

The FETCh and MEASure instructions also read various parameter values, except that FETCh reads the last measured value and MEASure is the re-measured value. FETCh speed faster, but the accuracy of MEASure higher.

Chapter 16 3-phase/series commands

Three-phase / series function commands is a special command of IT-M7721/ IT-M7722/IT-M7723E/IT-M7722D/IT-M7722E/IT-M7723D, but it's not suitable for IT-M7721L/IT-M7722L/IT-M7723/IT-M7723P.

Slave parameters are synchronized to the host, so the commands in this section are written or read by the communication card of the host (A). It is forbidden to write or read from the slave's communication card to avoid system operation error.

[SOURce:]NORMal:LINK

This command is used to connect / cancel two series units or three three-phase units.

Syntax

[SOURce:]NORMal:LINK

Arguments

None

Query Syntax

[SOURce:]NORMal:LINK?

Returns

<CONNECTED|DISCONNECTED>

Note: The response time of this command is at least greater than 1 second.

The following three-phase / series function commands apply:

- All machines must be successfully connected, that is, the [LINK] indicator lights.
- Without parameter A | B | C | ALL, only master A is set.

[SOURce:]NORMal:PHASe:DEGree

This command is used to set the angle difference of the three phases.

- In 3-phase mode, phase A is the reference, the default is 0 degree and

cannot be changed. And the phase difference between each two power supplies cannot be less than 5 degree.

- In series mode, phase A is the reference and defaults to 0 degree, Phase B is 180 degree and neither can be changed.

Syntax

[SOURce:]NORMal:PHASe:DEGree <A|B|C>,<NRf>

Arguments

<A|B|C> phase, <NRf> phase difference

Query Syntax

[SOURce:]NORMal:PHASe:DEGree? <A|B|C|ALL>

Returns

<NRf|NRf|NRf|NRf,NRf,NRf>

[SOURce:]NORMal:VOLTage:AC[:LEVel][:IMMediate][:AMPLitude]

This command sets the AC voltage of the power supply.

Syntax

[SOURce:]NORMal:VOLTage:AC[:LEVel][:IMMediate][:AMPLitude]

<A|B|C|ALL>,<NRf>

Arguments

<A|B|C|ALL> phase, <NRf> AC voltage

Query Syntax

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

<A|B|C|ALL>

Returns

<NRf|NRf|NRf|NRf,NRf,NRf>

[SOURce:]NORMal:FREQuency[:LEVel][:IMMediate]

This command sets the AC frequency of the power supply. This command can

synchronize all frequencies.

Syntax

[SOURce:]NORMal:FREQuency[:LEVel][:IMMediate] <NRf>

Arguments

<NRf> AC frequency

Query Syntax

[SOURce:]NORMal:FREQuency[:LEVel][:IMMediate]?

Returns

<NRf>

[SOURce:]OUTPut[:STATe]

This command sets the output state of the power supply. This command can synchronize all output or close.

Syntax

[SOURce:]OUTPut[:STATe] <0|1|OFF|ON>

Arguments

<0|1|OFF|ON> output state

Query Syntax

[SOURce:]OUTPut[:STATe]?

Returns

<0|1|OFF|ON>

PROTect:RMS:VOLTage

This command is used to set the OVPrms value.

Syntax

PROTect:RMS:VOLTage <A|B|C|ALL>, <NRf>

Arguments

<A|B|C|ALL>,<NRf>

Query Syntax

PROTect:RMS:VOLTage? <A|B|C|ALL>

Returns

<NRf|NRf|NRf|NRf,NRf,NRf>

PROTect:PEAK:VOLTage

This command is used to set the OVPpeak value.

Syntax

PROTect:PEAK:VOLTage <A|B|C|ALL>,<NRf>

Arguments

<A|B|C|ALL>,<NRf>

Query Syntax

PROTect:PEAK:VOLTage? <A|B|C|ALL>

Returns

<NRf|NRf|NRf|NRf,NRf,NRf>

PROTect:RMS:UNVOLTage

This command is used to set the UVPrms value.

Syntax

PROTect:RMS:UNVOLTage <A|B|C|ALL>,<NRf>

Arguments

<A|B|C|ALL>,<NRf>

Query Syntax

PROTect:RMS:UNVOLTage? <A|B|C|ALL>

Returns

<NRf|NRf|NRf|NRf,NRf,NRf>

PROTect:RMS:CURRent

This command is used to set the OCPrms value.

Syntax

PROTect:RMS:CURRent <A|B|C|ALL>,<NRf>

Arguments

<A|B|C|ALL>,<NRf>

Query Syntax

PROTect:RMS:CURRent? <A|B|C|ALL>

Returns

<NRf|NRf|NRf|NRf,NRf,NRf>

PROTect:PEAK:CURRent

This command is used to set the OCPpeak value.

Syntax

PROTect:PEAK:CURRent <A|B|C|ALL>,<NRf>

Arguments

<A|B|C|ALL>,<NRf>

Query Syntax

PROTect:PEAK:CURRent? <A|B|C|ALL>

Returns

<NRf|NRf|NRf|NRf,NRf,NRf>

PROTect:RMS:CURRent:TIME

This command is used to set the delay time of OCP.

Syntax

PROTect:RMS:CURRent:TIME <A|B|C|ALL>,<NR1>

Arguments

<A|B|C|ALL>,<NR1> 0 ~ 9999 ms

Query Syntax

PROTect:RMS:CURRent:TIME? <A|B|C|ALL>

Returns

<NRf|NRf|NRf|NRf,NRf,NRf>

PROTect:POWer

This command is used to set the delay time of OCP.

Syntax

PROTect:POWer <A|B|C|ALL>,<NRf>

Arguments

<A|B|C|ALL>,<NRf>

Query Syntax

PROTect:POWer? <A|B|C|ALL>

Returns

<NRf|NRf|NRf|NRf,NRf,NRf>

PROTect:MAX:CURRent:LIMit

This command sets the max current limit of the power supply.

Syntax

PROTect:MAX:CURRent:LIMit <A|B|C|ALL>,<NRf>

Arguments

<A|B|C|ALL>,<NRf>

Query Syntax

PROTect:MAX:CURRent:LIMit? <A|B|C|ALL>

Returns

<NRf|NRf|NRf|NRf,NRf,NRf>

[SOURce:]NORMal:VOLTage:AC:MAX[:LEVel]

This command sets the maximum AC voltage of the power supply.

Syntax

[SOURce:]NORMal:VOLTage:AC:MAX[:LEVel] <A|B|C|ALL>,<NRf>

Arguments

<A|B|C|ALL> phase,<NRf> the maximum AC voltage

Query Syntax

[SOURce:]NORMal:VOLTage:AC:MAX[:LEVel]? <A|B|C|ALL>

Returns

<NRf|NRf|NRf|NRf,NRf,NRf>

[SOURce:]NORMal:VOLTage:AC:MIN[:LEVel]

This command sets the minimum AC voltage of the power supply.

Syntax

[SOURce:]NORMal:VOLTage:AC:MIN[:LEVel] <A|B|C|ALL>,<NRf>

Arguments

<A|B|C|ALL> phase,<NRf> the minimum AC voltage

Query Syntax

[SOURce:]NORMal:VOLTage:AC:MIN[:LEVel]? <A|B|C|ALL>

Returns

<NRf|NRf|NRf|NRf,NRf,NRf>

PROTect:RMS:CURRent:MAX[:LEVel]

This command is used to set the maximum value of the OCPrms setting value.

Syntax

PROTect:RMS:CURRent:MAX[:LEVel] <A|B|C|ALL>,<NRf>

Arguments

<A|B|C|ALL>,<NRf>

Query Syntax

PROTect:RMS:CURRent:MAX[:LEVel]? <A|B|C|ALL>

Returns

<NRf|NRf|NRf|NRf,NRf,NRf>

PROTect:RMS:CURRent:MIN[:LEVel]

This command is used to set the minimum value of the OCPrms setting value.

Syntax

PROTect:RMS:CURRent:MIN[:LEVel] <A|B|C|ALL>,<NRf>

Arguments

<A|B|C|ALL>,<NRf>

Query Syntax

PROTect:RMS:CURRent:MIN[:LEVel]? <A|B|C|ALL>

Returns

<NRf|NRf|NRf|NRf,NRf,NRf>

[SOURce:]NORMal:CURRent:RANGe

This command is used to set the current range of the power supply.

Syntax

[SOURce:]NORMal:CURRent:RANGe <A|B|C|ALL>,<0|1|2|AUTO|HIGH|LOW>

Arguments

<A|B|C|ALL>,<0|1|2|AUTO|HIGH|LOW>

Query Syntax

[SOURce:]NORMal:CURRent:RANGe? <A|B|C|ALL>

Returns

<0|1|2|AUTO|HIGH|LOW>

[SOURce:]RELAy:MODe

This command is used to set the relay mode.

Syntax

[SOURce:]RELAy:MODe <A|B|C|ALL>,<0|1|OUTSYN|NC>

Arguments

<A|B|C|ALL>,<0|1|OUTSYN|NC>

OUTSYN: The relay is linked with Output; NC:The relay is normally closed. Whether

Query Syntax

[SOURce:]RELAy:MODe? <A|B|C|ALL>,

Returns

<OUTSYN|NC>

PROTect:CLEar

This command clears the protection status.

Syntax

PROTect:CLEar <A|B|C|ALL>

Arguments

<A|B|C|ALL>

FETCh[:SCALar]:VOLTage:AC?

This command reads the Vrms value.

Syntax

FETCh[:SCALar]:VOLTage:AC? <A|B|C|ALL>

Arguments

<A|B|C|ALL>

Returns

<NRf|NRf|NRf|NRf,NRf,NRf>

FETCh[:SCALar]:CURRent:AC?

This command reads the Irms value.

Syntax

FETCh[:SCALar]:CURRent:AC? <A|B|C|ALL>

Arguments

<A|B|C|ALL>

Returns

<NRf|NRf|NRf|NRf,NRf,NRf>

FETCh[:SCALar]:POWer[:REAL]?

This command reads the active power.

Syntax

FETCh[:SCALar]:POWer [:REAL]? <A|B|C|ALL>

Arguments

<A|B|C|ALL>

Returns

<NRf|NRf|NRf|NRf,NRf,NRf>

FETCh[:SCALar]:POWer:APParent?

This command reads the apparent power.

Syntax

FETCh[:SCALar]:POWer:APParent? <A|B|C|ALL>

Arguments

<A|B|C|ALL>

Returns

<NRf|NRf|NRf|NRf,NRf,NRf>

FETCh[:SCALar]:POWer:PFACtor?

This command reads the power factor.

Syntax

FETCh[:SCALar]:POWer:PFACtor? <A|B|C|ALL>

Arguments

<A|B|C|ALL>

Returns

<NRf|NRf|NRf|NRf,NRf,NRf>

FETCh[:SCALar]:FREQuency?

This command reads the frequency.

Syntax

FETCh[:SCALar]:FREQuency?

Arguments

None

Returns

< NRf >

FETCh[:SCALar]:CURRent:PEAK?

This command reads the positive peak current.

Syntax

FETCh[:SCALar]:CURRent:PEAK? <A|B|C|ALL>

Arguments

<A|B|C|ALL>

Returns

<NRf|NRf|NRf|NRf,NRf,NRf>

FETCh[:SCALar]:VOLTage:PEAK?

This command reads the positive peak voltage.

Syntax

FETCh[:SCALar]:VOLTage:PEAK? <A|B|C|ALL>

Arguments

<A|B|C|ALL>

Returns

<NRf|NRf|NRf|NRf,NRf,NRf>

FETCh[:SCALar]:CURRent:PEAK:PLUS?

This command reads the positive plus peak current.

Syntax

FETCh[:SCALar]:CURRent:PEAK:PLUS? <A|B|C|ALL>

Arguments

<A|B|C|ALL>

Returns

<NRf|NRf|NRf|NRf,NRf,NRf>

FETCh[:SCALar]:CURRent:PEAK:MINUs?

This command reads the positive minus peak current.

Syntax

FETCh[:SCALar]:CURRent:PEAK:MINUs? <A|B|C|ALL>

Arguments

<A|B|C|ALL>

Returns

<NRf|NRf|NRf|NRf,NRf,NRf>

FETCh[:SCALar]:THD?

This command reads the total voltage harmonic distortion.

Syntax

FETCh[:SCALar]:THD? <A|B|C|ALL>

Arguments

<A|B|C|ALL>

Returns

<NRf|NRf|NRf|NRf,NRf,NRf>

FETCh[:SCALar]:CURRent:THD?

This command reads the total current harmonic distortion.

Syntax

FETCh[:SCALar]:CURRent:THD? <A|B|C|ALL>

Arguments

<A|B|C|ALL>

Returns

<NRf|NRf|NRf|NRf,NRf,NRf>

FETCh[:SCALar]:POWer:REACTive?

This command reads the reactive power.

Syntax

FETCh[:SCALar]:POWer:REACTive? <A|B|C|ALL>

Arguments

<A|B|C|ALL>

Returns

<NRf|NRf|NRf|NRf,NRf,NRf>

MEASure[:SCALar]:VOLTage:AC?

This command reads the Vrms value.

Syntax

MEASure[:SCALar]:VOLTage:AC? <A|B|C|ALL>

Arguments

<A|B|C|ALL>

Returns

<NRf|NRf|NRf|NRf,NRf,NRf>

MEASure[:SCALar]:CURRent:AC?

This command reads the Irms value.

Syntax

MEASure[:SCALar]:CURRent:AC? <A|B|C|ALL>

Arguments

<A|B|C|ALL>

Returns

<NRf|NRf|NRf|NRf,NRf,NRf>

MEASure[:SCALar]:POWer[:REAL]?

This command reads the active power.

Syntax

MEASure[:SCALar]:POWer[:REAL]? <A|B|C|ALL>

Arguments

<A|B|C|ALL>

Returns

<NRf|NRf|NRf|NRf,NRf,NRf>

MEASure[:SCALar]:POWer:APParent?

This command reads the apparent power.

Syntax

MEASure[:SCALar]:POWer:APParent? <A|B|C|ALL>

Arguments

<A|B|C|ALL>

Returns

<NRf|NRf|NRf|NRf,NRf,NRf>

MEASure[:SCALar]:POWer:PFACtor?

This command reads the power factor.

Syntax

MEASure[:SCALar]:POWer:PFACtor? <A|B|C|ALL>

Arguments

<A|B|C|ALL>

Returns

<NRf|NRf|NRf|NRf,NRf,NRf>

MEASure[:SCALar]:FREQuency?

This command reads the frequency.

Syntax

MEASure[:SCALar]:FREQuency?

Arguments

None

Returns

< NRf >

MEASure[:SCALar]:CURRent:PEAK?

This command reads the positive peak current.

Syntax

MEASure[:SCALar]:CURRent:PEAK? <A|B|C|ALL>

Arguments

<A|B|C|ALL>

Returns

<NRf|NRf|NRf|NRf,NRf,NRf>

MEASure[:SCALar]:VOLTage:PEAK?

This command reads the positive peak voltage.

Syntax

MEASure[:SCALar]:VOLTage:PEAK? <A|B|C|ALL>

Arguments

<A|B|C|ALL>

Returns

<NRf|NRf|NRf|NRf,NRf,NRf>

MEASure[:SCALar]:CURRent:PEAK:PLUS?

This command reads the positive plus peak current.

Syntax

MEASure[:SCALar]:CURRent:PEAK:PLUS? <A|B|C|ALL>

Arguments

<A|B|C|ALL>

Returns

<NRf|NRf|NRf|NRf,NRf,NRf>

MEASure[:SCALar]:CURRent:PEAK:MINUs?

This command reads the positive minus peak current.

Syntax

MEASure[:SCALar]:CURRent:PEAK:MINUs? <A|B|C|ALL>

Arguments

<A|B|C|ALL>

Returns

<NRf|NRf|NRf|NRf,NRf,NRf>

MEASure[:SCALar]:THD?

This command reads the total voltage harmonic distortion.

Syntax

MEASure[:SCALar]:THD? <A|B|C|ALL>

Arguments

<A|B|C|ALL>

Returns

<NRf|NRf|NRf|NRf,NRf,NRf>

MEASure[:SCALar]:CURR:THD?

This command reads the total current harmonic distortion.

Syntax

MEASure[:SCALar]:CURR:THD? <A|B|C|ALL>

Arguments

<A|B|C|ALL>

Returns

<NRf|NRf|NRf|NRf,NRf,NRf>

MEASure[:SCALar]:POWer:REACTive?

This command reads the reactive power.

Syntax

MEASure[:SCALar]:POWer:REACTive? <A|B|C|ALL>

Arguments

<A|B|C|ALL>

Returns

<NRf|NRf|NRf|NRf,NRf,NRf>

MEASure?

This command is used to re-read the values of Arguments such as Vrms, Vdc, Irms, Idc, Ipk_plus, Ipk_minus, W (Real Power), PF, IPmax, VA, Q(VAR), VTHD, Freq, Vpeak, VAC, IAC, ITHD.

Syntax

MEASure? <A|B|C>

Arguments

<A|B|C>

Returns

< Vrms,Vdc,Irms,Idc,Ipk_plus,Ipk_minus,W(Real Power),PF,IPmax,VA,Q(VAR),VTHD,Freq,Vpeak,VAC,IAC,ITHD>

FETCh?

This command is used to read the values of Arguments such as Vrms, Vdc, Irms, Idc, Ipk_plus, Ipk_minus, W (Real Power), PF, IPmax, VA, Q(VAR), VTHD, Freq, Vpeak, VAC, IAC, ITHD.

Syntax

FETCh? <A|B|C>

Arguments

<A|B|C>

Returns

< Vrms,Vdc,Irms,Idc,Ipk_plus,Ipk_minus,W(Real Power),PF,IPmax,VA,Q(VAR),VTHD,Freq,Vpeak,VAC,IAC,ITHD>

Chapter 17 RS485 Communication Description

Command Format

Head	DA	SA	Data
0xBA	Destination Address	Source Address	SCPI command

Description:

- Head: Head string 0XBA (the 0xBA is fixed).
- DA: Destination Address (broadcast command using 0x7F = 127).

  PS: Broadcast is only for setting command not for query command.
- SA: Source Address(The source Address of the upper machine, which cannot be the same as the machine address.)
- Data: SCPI command.

Example

1. Set the source and destination addresses.

   a) The source address of the upper machine is: 0x02 = 2.

   b) The destination address of the IT-M7700 machine is set to: 0x10 = 16 (Panel Operation System > 4:I/O Advance Config > 2:485 Address 16)

2. Send the command to ask for the output status: OUTP?

Head	DA	SA	O	U	T	P	?	CR	LF
0xBA	0x10	0x02	0x4F	0x55	0x54	0x50	0x3F	0x0D	0x0A

3. Check the IT-M7700 return value.

Head	DA	SA	O	F	F	CR	LF
0xBA	0x02	0x10	0x4F	0x46	0x46	0x0D	0x0A

Chapter 18 Programming Examples

This chapter displays the programming examples to remotely control IT-M7700 power supply using SCPI commands.

Note

- If the user want to change the settings of the instrument, for instance, the output setting value, the command SYST:REM must be sent to the instrument after finishing the connection between the instrument and PC.
- " - >" indicates the commands that you send to the IT-M7700 power supply.
- Self-defined Waveform, Dimmer, Surge/Trap and List commands is a special command of IT-M7721/IT-M7722/IT-M7723/IT-M7723E/IT-M7722D/IT-M7722E/IT-M7723D, but it's not suitable for IT-M7721L/IT-M7722L.

Example 1: Identifying the Power Supply in Use

You can verify whether you are communicating with the right IT-M7700 power supply.

To query the identification of the power supply, send the command:

```
-> *IDN?
```

To check the power supply error queue, send the command:

```
-> SYST:ERR?
```

Example 2: Applying DC Output

The IT-M7700 is configured as a DC power supply and outputs 20V voltage. Please enter the following command:

```
-> SYSTem:REMote
-> NORMal:MODE DC
-> NORMal:VOLTage:DC 20.0
-> PROTect:MAX:CURRent:LIMit 20.0
-> OUTPut ON
-> MEASure:VOLTage:DC?
-> MEASure:CURRent:DC?
-> MEASure:POWer?
```

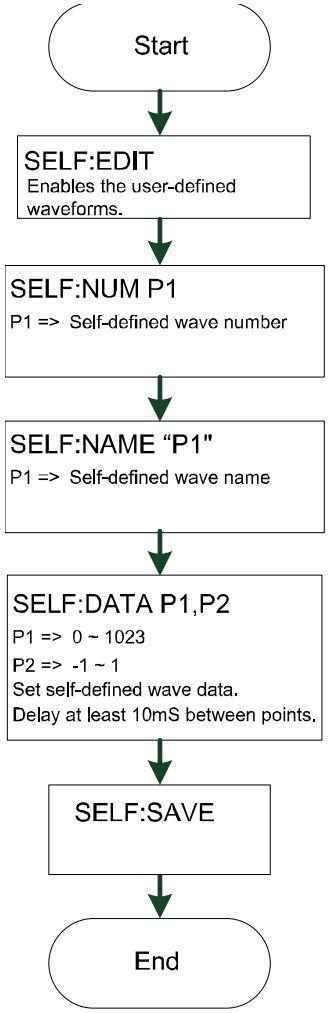
Example 3: Applying Waveform Output

The ITM7700 is configured as a waveform generator and outputs 10V/50Hz Sine wave. Please enter the following command:

```
-> SYSTem:REMote
-> NORMal:MODE AC
-> NORMal:VOLTage:AC 10.0
-> NORMal:FREQuency 50.0
-> NORMal:PHASe:STARt 45.0
-> NORMal:PHASe:STOP 0.0
-> NORMal:WAVE SINE
-> PROTect:MAX:CURRent:LIMit 20.0
-> OUTPut ON
-> MEASure:VOLTage:AC?
-> MEASure:CURRent:AC?
-> MEASure:POWer?
-> MEASure:POWer:APParent?
-> MEASure:POWer:PFACtor?
-> MEASure:FREQuency?
-> MEASure:THD?
-> MEASure:POWer:REACTive?
```

Example 4: Self-defined Waveform

Users can self define arbitrary waveforms to simulate or duplicate the real waveforms. The execution flowchart of Self-defined Waveform is as follows.



Please enter the following command:

```
-> SYSTem:REMote

-> SELFdefine:EDIT

-> SELFdefine:NUMber 1

-> SELFdefine:NAME "Selfuser1"

-> SELFdefine:DATA 0,0.5

-> SELFdefine:DATA 1,0.4

……..

-> SELFdefine:SAVE
```

Example 5: Dimmer Function

The Dimmer Function aims to adjust the light illuminance intensity by setting the phase angle and concealing the leading edge or trailing edge waveform.

Please enter the following command:

```
-> SYSTem:REMote
-> NORMal:MODE AC
-> NORMal:VOLTage:AC 10.0
-> NORMal:FREQuency 50.0
-> NORMal:WAVE SINE
-> NORMal:DIMMer:MODe 1
-> NORMal:DIMMer 90°
-> OUTPut 1
```

Example 6: Surge/Trap Function

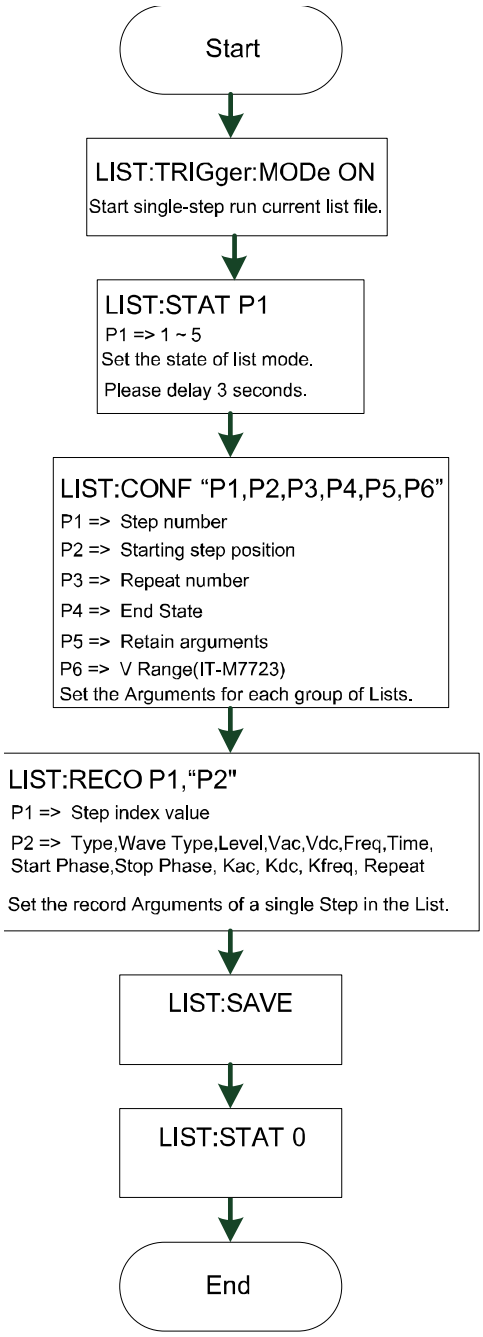
IT-M7700 provides surge/trap simulation. The user can add surge/trap to simulate abnormal voltage fluctuation on the basis of outputting Sin wave, and test usage of the DUT under this circumstance. Please enter the following command:

```
-> SYSTem:REMote
-> NORMal:MODE AC
-> NORMal:VOLTage:AC 10.0
-> NORMal:FREQuency 50.0
-> NORMal:WAVE SINE
-> NORMal:SURGETRAP:MODe 1
-> NORMal:SURGETRAP:PERIOD 2
-> NORMal:SURGETRAP:WIDTH 1
-> NORMal:SURGETRAP:PERCENT 200%
-> OUTPut 1
```

Example 7: List Function

Edit List

You can use the list program to perform a variety of output waveform simulations. The execution flowchart of list program is as follows.



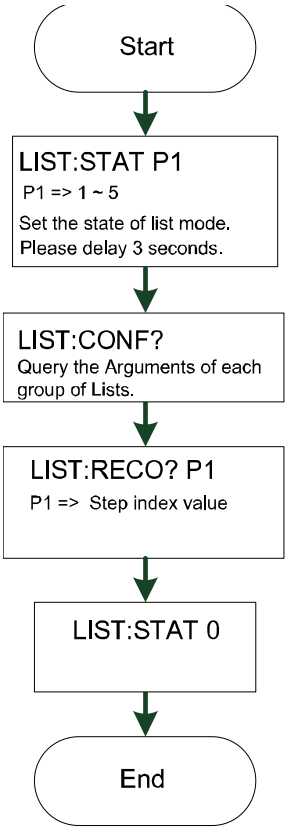
Please enter the following command:

```
-> SYSTem:REMote

-> LIST:TRIGger:MODe ON

-> LIST:STATe 1

-> LIST:CONFigure "50,1,2,0,0,0" (IT-M7723)

-> LIST:CONFigure "50,1,2,0,0" (Other models except 7723.)

-> LIST:RECOrder
1,"0,1,100.0,110.0,0.0,50.0,1,0.0,180.0,1,1,1,2"
```

```
->LIST:RECOrder
2,"0,0,100.0,100.0,0.0,50.0,1,0.0,180.0,1,1,1,2"
->LIST:RECOrder
3,"0,2,100.0,90.0,0.0,50.0,1,0.0,180.0,1,1,1,2"
->LIST:RECOrder
4,"0,3,100.0,80.0,0.0,50.0,1,0.0,180.0,1,1,1,2"
->LIST:RECOrder
5,"0,4,10.0,70.0,0.0,50.0,1,0.0,180.0,1,1,1,2"
-> LIST:SAVE 1
-> LIST:STATe 0
```

View List File

Before selecting and running the list program, you can view the related parameters and each step of the existed list program. The operation steps to view the list program are as follows.



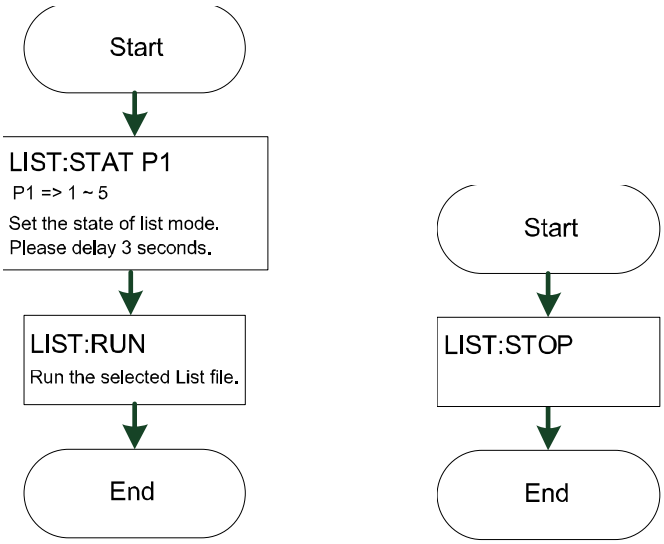
Please enter the following command:

```
-> SYSTem:REMote
-> LIST:STATe 1
```

```
-> LIST:CONFigure?
-> LIST:RECOrder? 1
-> LIST:RECOrder? 2
-> LIST:RECOrder? 3
-> LIST:RECOrder? 4
-> LIST:RECOrder? 5
-> LIST:STATe 0
```

Run/Stop List

The user can run the existed list program as needed to enable outputting of corresponding waveform sequence by the power supply. Detailed operation steps are as below:



Please enter the following command:

```
-> SYSTem:REMote
-> LIST:STATe 1
-> LIST:RUN
-> LIST:STOP
```

Example 8: THD waveform(IT-M7723P)

```
->SYST:REM
->SELF:USER:INDEX 0
```

```
->SELF:USER:TYPE THD
->SELF:USER:THD:METHOD THDF
->SELF:USER:THD:DATA 2,2.6,45
->SELF:USER:THD:DATA 3,3.06,46
->SELF:USER:THD:DATA 4,4.63,52
…….

->SELF:USER:THD:DATA 49,3.20,86.6
->SELF:USER:THD:DATA 50,2.6,121.0
->SELF:USER:THD:SAVE
/*  Delay 500ms */
->SELF:USER:THD:RECALL 0
/*  Delay 200ms */
->SELF:USER:THD:DATA? 2
->SELF:USER:THD:DATA? 3
->SELF:USER:THD:DATA? 4
…….

->SELF:USER:THD:DATA? 50
```

Example 9: Self-defined Waveform(IT-M7723P)

```
->SYST:REM
->SELF:USER:INDEX 1
->SELF:USER:TYPE POINT
->SELF:USER:POINT:METHOD POINTS
->SELF:USER:POINT:LEN 11
->SELF:USER:POINT:DATA 0,-1
->SELF:USER:POINT:DATA 102,-0.8
->SELF:USER:POINT:DATA 205,-0.6
->SELF:USER:POINT:DATA 307,-0.4
->SELF:USER:POINT:DATA 410,-0.2
```

```
->SELF:USER:POINT:DATA 512,0.0
->SELF:USER:POINT:DATA 614,0.2
->SELF:USER:POINT:DATA 717,0.4
->SELF:USER:POINT:DATA 819,0.6
->SELF:USER:POINT:DATA 921,0.8
->SELF:USER:POINT:DATA 1023,1
->SELF:USER:POINT:SAVE
/* Delay 4000ms */
->SELF:USER:POINT:RECALL 1
/* Delay (11 x 4 )ms   */
->SELF:USER:POINT:DATA? 0
->SELF:USER:POINT:DATA? 1
->SELF:USER:POINT:DATA? 2
->SELF:USER:POINT:DATA? 3
…….

->SELF:USER:POINT:DATA? 9
->SELF:USER:POINT:DATA? 10
```

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.