



Series 4000

LDC40xx

TED4015

ITC40xx

SCPI Programmer's Reference Manual



2015

Version: 3.3
Date: 08-Apr-2015

Contents

Foreword	4
1 General Information	5
2 Introduction to the SCPI Language	6
2.1 Syntax Conventions	6
2.2 Command Separators	7
2.3 Using the MIN, MAX and DEF Parameters	7
2.4 Querying Parameter Settings	8
2.5 Command Terminators	8
2.6 Parameter Types	8
2.6.1 Numeric Parameters	8
2.6.2 Discrete Parameters	9
2.6.3 Boolean Parameters	9
2.6.4 ASCII String Parameters	9
3 Commands by Subsystem	10
3.1 IEEE-488.2 Commands	10
3.1.1 *IDN? - Identification Query	11
3.1.2 *RST - Reset	11
3.1.3 *SAV, *RCL, *SDS - Save/Recall Setup	12
3.1.4 *TST? - Self Test	12
3.1.5 *OPC, *OPC? - Operation Complete	12
3.1.6 *WAI - Wait To Continue	13
3.1.7 *STB?	13
3.1.8 *SRE, *SRE?	14
3.1.9 *CLS	14
3.1.10 *ESR?, *ESE, *ESE?	14
3.2 System Subsystem Commands	16
3.2.1 Beeper	16
3.2.2 Error Query	16
3.2.3 Line Frequency	17
3.2.4 SCPI Version Query	17
3.3 Measurement Commands	18
3.3.1 Abort Measurement in Progress	22
3.3.2 Configuring a Measurement	22
3.3.3 Initiating a Measurement	24
3.3.4 Fetch a Measurement	24
3.3.5 Read Measurement Data	26
3.3.6 Simple Measurement	26
3.4 LD Output Commands	28
3.4.1 Output State	29

3.4.2	Switch-On Delay _____	29
3.4.3	Polarity _____	30
3.4.4	Noise Reduction Filter _____	30
3.4.5	Protection Voltage _____	31
3.4.6	Temperature Protection Mode _____	31
3.4.7	LD-ENABLE Input Mode _____	32
3.4.8	Protection Queries _____	33
3.5	PD Input Commands _____	34
3.5.1	Polarity _____	34
3.5.2	BIAS State _____	35
3.5.3	BIAS Voltage _____	35
3.5.4	Input Routing _____	36
3.6	Thermopile/Power Meter Input Commands _____	37
3.6.1	Input Routing _____	37
3.7	LD Source Commands _____	38
3.7.1	LD Source Function _____	40
3.7.2	LD Current Limit _____	40
3.7.3	LD Current Setpoint _____	41
3.7.4	LD Optical Power _____	41
3.7.5	LD Power Feedback _____	43
3.7.6	LD Amplitude Modulation _____	43
3.7.7	QCW Pulse Setting _____	44
3.8	PD Sense Commands _____	45
3.8.1	Photodiode Current Range _____	45
3.8.2	Photodiode Current Protection _____	46
3.8.3	Photodiode Responsivity _____	46
3.8.4	Photodiode Power Range _____	47
3.8.5	Photodiode Power Protection _____	47
3.9	Thermopile/Power Meter Sense Commands _____	47
3.9.1	Thermopile/Power Meter Voltage Range _____	48
3.9.2	Thermopile/Power Meter Voltage Protection _____	49
3.9.3	Thermopile/Power Meter Responsivity _____	49
3.9.4	Thermopile/Power Meter Power Range _____	50
3.9.5	Thermopile/Power Meter Power Protection _____	50
3.10	QCW Pulse Trigger Commands _____	51
3.10.1	QCW Trigger Source _____	51
3.11	TEC Source Commands _____	52
3.11.1	TEC Source Function _____	53
3.11.2	TEC Current Limit _____	54
3.11.3	TEC Current Setpoint _____	54
3.11.4	TEC Temperature Setpoint Limit _____	55
3.11.5	TEC Temperature Setpoint _____	55
3.11.6	PID Loop Constants _____	56

3.11.7	PID Auto Tune	56
3.12	Temperature Sense Commands	58
3.12.1	Sensor Selection	60
3.12.2	Thermistor Calculations	60
3.12.3	Transducer Offset	61
3.12.4	Temperature Protection	62
3.13	TEC Output Commands	63
3.13.1	Output State	63
3.13.2	Protection Queries	64
3.14	Status Subsystem Commands	65
3.14.1	Programming and Reading Registers	68
3.14.2	Auxiliary Register Group	69
3.14.3	Measurement Register Group	70
3.14.4	Questionable Register Group	71
3.14.5	Operation Registers Group	72
3.14.6	Status Preset Command	73
3.15	Memory Subsystem Commands	74
3.15.1	Number of State Memories	74
3.15.2	State Memories Names	74
3.16	Display Subsystem Commands	75
3.16.1	Display Brightness	75
3.16.2	Display Contrast	75
3.17	Digital I/O Output Commands	76
3.17.1	Digital I/O Output	76
3.17.2	Digital I/O Configuration	77
3.18	Digital I/O Input Commands	77
3.18.1	Digital I/O Input	78
3.19	Unit Subsystem Commands	79
3.19.1	Temperature Unit Commands	79
3.20	Calibration Subsystem Commands	80
3.20.1	Calibration String	80
4	SCPI Error Messages	81
4.1	Command Errors	82
4.2	Execution Errors	83
4.3	Device Specific Errors	84
4.4	Query Errors	84
4.5	Instrument Errors	85
5	Appendix	86
5.1	Instrument Default Settings	86
5.2	Copyright and Exclusion of Reliability	91
5.3	Thorlabs Worldwide Contacts	92

We aim to develop and produce the best solution for your application in the field of optical measurement technique. To help us to live up to your expectations and improve our products permanently we need your ideas and suggestions. Therefore, please let us know about possible criticism or ideas. We and our international partners are looking forward to hearing from you.

Thorlabs GmbH

Warning

Sections marked by this symbol explain dangers that might result in personal injury or death. Always read the associated information carefully, before performing the indicated procedure.

Attention

Paragraphs preceded by this symbol explain hazards that could damage the instrument and the connected equipment or may cause loss of data.

Note

This manual also contains "NOTES" and "HINTS" written in this form.

Please read these advices carefully!

1 General Information

This part of the programmer's reference manual contains specific information about the SCPI command set of the Series 4000 instruments. A general description is followed by explanations of how to use the SCPI commands. The command set is according to the SCPI 1999.0 standard.

The Series 4000 instruments provide a USB 2.0 Full Speed link according to the USB 2.0 specification, the USBTMC specification and the USBTMC USB488 specification.

The instrument's command buffer accepts commands with a maximum length of 255 characters. The instrument's message output buffer can contain response messages of up to 255 characters.

There are VXIpnnp/VISA instrument drivers available for Series 4000 instruments. See www.thorlabs.com for downloads.

Related documents:

- TED4000 Series Operation Manual available at www.thorlabs.com
- LDC4000 Series Operation Manual available at www.thorlabs.com
- ITC4000 Series Operation Manual available at www.thorlabs.com
- USB Test and Measurement Class (USBTMC) Specification, Revision 1.0, available at www.usb.org
- USB Test and Measurement Class, Subclass USB488 Specification (USBTMC USB488), Revision 1.0, available at www.usb.org
- IEEE Std 488.2-1992, IEEE Standard Codes, Formats, Protocols, and Common Commands, available at www.ieee.org
- SCPI Specification, Version 1999.0, May, 1999, available at www.ivifoundation.org

2 Introduction to the SCPI Language

SCPI (Standard Commands for Programmable Instruments) is an ASCII based instrument command language designed for test and measurement instruments, based on a hierarchical tree structure. It means associated commands are grouped together under a common node. For example:

```
SOURce[1]
  :CURRent
    :LIMit
      [:AMPLitude] {MIN|MAX|<amps>}
```

`SOURce` is the root level keyword of the command, `CURRent` is the second level keyword, `LIMit` is the third level keyword and `AMPLitude` is the fourth level keyword, which can be omitted in this case. A colon (:) separates between command levels.

For further information about SCPI please see SCPI Standard available at www.ivifoundation.org.

2.1 Syntax Conventions

The format used in this manual for commands is illustrated below:

```
SOURce[1]:CURRent {MINimum|MAXimum|<amps>}
```

The command syntax shows most commands (and some parameters) as a mixture of upper- and lower-case letters. The upper-case letters indicate the abbreviated spelling for the command. For shorter program lines, you can use the abbreviated form. For better program readability, you can send the long form. In the above example `CURR` and `CURRent` are both acceptable syntax spellings. Other spellings like `CUR` or `CURRE` are not allowed and will generate an error. Commands are not case sensitive.

Curly braces ({}) enclose the parameter choices for a given command string. Braces are not sent with the command string.

A vertical bar (|) separates multiple parameter choices for a given command string. For example, `{MINimum|MAXimum|<amps>}` in the above command indicates that you can specify a numeric current parameter, or `MINimum`, or `MAXimum`. The bar is not sent with the command string.

Triangle brackets (<>) indicate that you must specify a value for the enclosed parameter. For example, the above syntax statement shows the `<amps>` parameter enclosed in triangle brackets. The brackets are not sent with the command string. You must specify a value for the parameter (for example `5.0`) unless you select one of the other options shown in the syntax (for example `MIN`).

Some parameters and even parts of the command itself are enclosed in square brackets (`[]`). This indicates that the part of the command or the parameter is optional and can be omitted. The brackets are not sent with the command string. If you do not specify a value for an optional parameter, the instrument chooses a default value.

2.2 Command Separators

A colon (:) is used to separate a command keyword from a lower-level keyword. You must insert a blank space to separate a parameter from a command keyword. If a command requires more than one parameter, you must separate adjacent parameters using a comma (,) as shown below:

```
MEM:STAT:NAME 1,"My setting"
```

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

```
DISP:BRIG 1;CONT 0.5
```

... is the same as sending the following two commands:

```
DISP:BRIG 1
```

```
DISP:CONT 0.5
```

Use a semicolon and a colon to link commands from different subsystems. The colon resets the command tree to the root level. For example:

```
SOURce:CURRent 0.5;;OUTPut ON
```

2.3 Using the MIN, MAX and DEF Parameters

For many commands, you can substitute `MINimum` or `MAXimum` in place of a parameter. In some cases you may also substitute `DEFault`.

For example, consider the following command:

```
SOURce:TEMPerature:PROTection:WINDow {MIN|MAX|DEF|<temp>}
```

```
SOURce[1]
```

```
  :CURRent
```

```
    :LIMit
```

```
      [:AMPLitude] {MIN|MAX|<amps>}
```

Instead of selecting a specific value for the `<temp>` parameter, you can substitute `MIN` or `MINimum` to set the value to its minimum, `MAX` or `MAXimum` to set the value to its maximum, `DEF` or `DEFault` to set the value to the default value.

For reasons of clarity the following parts of commands are written in short form only throughout this manual:

Original Form	Short Form
{MINimum MAXimum DEFault <value>}	{MIN MAX DEF <value>}
{MINimum MAXimum DEFault}	{MIN MAX DEF}
{MINimum MAXimum <value>}	{MIN MAX <value>}
{MINimum MAXimum}	{MIN MAX}

2.4 Querying Parameter Settings

You can query the current value of most parameters by adding a question mark (?) to the command. For example, the following command sets the temperature setpoint:

```
SOURce:CURRent 5.0
```

You can then query the value by typing:

```
SOURce:CURRent?
```

You can also query the minimum or maximum and - where specified - the default values as follows:

```
SOURce:CURRent? MINimum
```

```
SOURce:CURRent? MAXimum
```

2.5 Command Terminators

A command string sent to the instrument must terminate with a <new line> (<NL>) character. Command string termination will always reset the current SCPI command path to the root level.

2.6 Parameter Types

The SCPI language defines several data formats to be used in program messages and response messages.

2.6.1 Numeric Parameters

Commands that require numeric parameters will accept all commonly used decimal representations of numbers including optional signs, decimal points, and scientific notation. Special values for numeric parameters such as `MIN`, `MAX`, and `DEF` may also be accepted. You can also send engineering unit suffixes with numeric parameters (e.g., M, k, m, or u). If a command accepts only certain specific values, the instrument will automatically round the input numeric parameter to the accepted values.

Additionally numeric parameters can be programmed using binary, decimal, hexadecimal or octal data formats for the parameter value (for a detailed description of numeric formats see chapter 7.7.4 <NONDECIMAL NUMERIC PROGRAM DATA> of IEEE488.2-1992).

Example

The four commands program the *Auxiliary Enable Register* to the same value:

- `STAT:AUX:ENAB 2081`
Program the *Auxiliary Enable Register* with 2018 decimal.
- `STAT:AUX:ENAB #H821`
Program the *Auxiliary Enable Register* with 821 hexadecimal.
- `STAT:AUX:ENAB #Q4041`
Program the *Auxiliary Enable Register* with 4041 octal.
- `STAT:AUX:ENAB #B100000100001`
Program the *Auxiliary Enable Register* with 100000100001 binary.

2.6.2 Discrete Parameters

Discrete parameters are used to program settings that have a limited number of values (like `CELSius`, `FAHRenheit` or `KELVin`). They have a short form and a long form just like command keywords. You can mix upper- and lower-case letters. Query responses will always return the short form in all upper-case letters.

Example

The following command requires a discrete parameter for the temperature unit:

```
→ UNIT:TEMPerature {C|CEL|CELSius|F|FAR|FAHRenheit|K|KELVin}
```

Discrete parameters may also have a default condition. In this manual the default conditions are underlined.

2.6.3 Boolean Parameters

Boolean parameters represent a single binary condition that is either true or false. For a false condition, the instrument will accept `OFF` or `0`. For a true condition, the instrument will accept `ON` or `1`. When you query a boolean setting, the instrument will always return `0` or `1`.

Boolean parameters may also have a default condition. In this manual the default conditions are underlined.

2.6.4 ASCII String Parameters

String parameters can contain virtually any set of ASCII characters. A string must begin and end with matching quotes; either with a single quote or a double quote. You can include the quote delimiter as part of the string by typing it twice without any characters in between.

3 Commands by Subsystem

3.1 IEEE-488.2 Commands

The IEEE-488.2 standard defines a set of common commands that perform functions such as reset, self-test, and status operations. Common commands always begin with an asterisk (*), are three characters in length, and may include one or more parameters. The command keyword is separated from the first parameter by a blank space. Use a semicolon (;) to separate multiple commands. IEEE488.2 commands are device commands that are common to all devices according to the IEEE488.2 standard. Most of the commands are described in detail in this section. The following common commands associated with the status structure are related to the [STATus Subsystem Commands](#)^[65] section: *CLS, *ESE, *ESE?, *ESR?, *SRE, *SRE?, *STB?

See also figure [Status Structure](#)^[65].

Table <%TABLENO++%>: IEEE-488.2 commands

Mnemonic	Name	Description
*CLS ^[14]	Clear status	Clear all event registers and Error Queue
*ESE ^[14] <value>	Event enable command	Program the <i>Standard Event Enable Register</i>
*ESE? ^[14]	Event enable query	Read the <i>Standard Event Enable Register</i>
*ESR? ^[14]	Event status register query	Read and clear the <i>Standard Event Register</i>
*IDN? ^[11]	Identification query	Read the unit's identification string
*OPC ^[12]	Operation complete command	Set the Operation Complete bit in the <i>Standard Event Register</i>
*OPC? ^[12]	Operation complete query	Place a 1 into the output queue when all device operations have been completed
*RCL ^[12] <value>	Recall command	Return the unit to the user-saved setup
*RST ^[11]	Reset command	Return the unit to the *RST default condition
*SAV ^[12] <value>	Save default device setting	Initialize the content of save/recall registers
*SDS ^[12] <value>	Save default device setting	Initialize the content of save/recall registers
*SRE ^[13] <value>	Service request enable command	Program the <i>Service Request Enable Register</i>
*SRE? ^[13]	Service request enable query	Read the <i>Service Request Enable Register</i>
*STB? ^[13]	Status byte query	Read the <i>Status Byte Register</i>
*TST? ^[12]	Self-test query	Perform the unit's self-test and return the result.
*WAI ^[13]	Wait-to-continue command	Wait until all previous commands are executed

3.1.1 *IDN? - Identification Query

Syntax

*IDN?

Description

The identification query reads the instrument's identification code which contains four comma-separated fields. The first field is the manufacturer, the second field is the model code, the third field is the serial number, and the fourth field is the firmware revision code which contains two (LDC4000) or three (TED4000, ITC4000) codes separated by slashes (/).

The query is sent in the following format:

```
THORLABS,MMM,SSS,X.X.X/Y.Y.Y/Z.Z.Z
```

Where:

MMM	is the model code
SSS	is the serial number
X.X.X	is the instrument firmware revision level
Y.Y.Y	is the firmware revision level of the front panel board
Z.Z.Z	is the firmware revision level of the temperature controller board (TED4000, ITC4000 only)

Example

→ *IDN?

Query the ID code.

← THORLABS,ITC4020,E12345678,1.4.0/2.0.3/1.6.0

Typical response.

3.1.2 *RST - Reset

Syntax

*RST

Description

When the *RST command is sent, the instrument performs the following operations:

- Switches off the instrument's outputs.
- Sets the direction of general purpose I/O1..4 to input.
- Switches photodiode BIAS to OFF

3.1.3 *SAV, *RCL, *SDS - Save/Recall Setup

Syntax

*SAV <value>

*RCL <value>

*SDS <value>

Description

Value: 0..7 Memory location 1 ... 8

Use the *SAV command to save the present instrument setup configuration in non-volatile (state-) memory for later recall. The *RCL command is used to restore the instrument to the saved setup configuration. Eight setup configurations can be saved and recalled.

Use the *SDS command to initialize the selected (state-) memory with factory default settings. Your settings will be overwritten. See also chapter [Memory Subsystem Commands](#)^[74] and appendix [Instrument Default Settings](#)^[86].

Example

→ *SAV 1

Saves the present instrument setup configuration to memory location 2.

3.1.4 *TST? - Self Test

Syntax

*TST?

Description

Use this query command to perform the instrument self-test routine. The command places the coded result in the output queue. A returned value of zero (0) indicates that the test passed, other values indicate that the test failed and the error code is placed into the instrument's error queue.

3.1.5 *OPC, *OPC? - Operation Complete

Syntax

*OPC

*OPC?

Description

When *OPC is sent, the OPC bit in the *Standard Event Register* will be set after all pending command operations are complete. When *OPC? is sent, an ASCII 1 is placed in the output queue after all pending command operations are complete.

Typically, either one of these commands is sent after the INITiate command. The INITiate command is used to take the instrument out of idle in order to perform measurements. While operating within the trigger model layers, many sent commands will not execute. After all programmed operations are completed, the instrument returns to the idle state at which time all pending commands (including *OPC and/or *OPC?) are executed. When executing the OPC bit in the status structure is set and/or an ASCII 1 is placed in the output queue.

3.1.6 *WAI - Wait To Continue

Syntax

*WAI

Description

The *WAI command is a no operation command for the instrument and thus, does not need to be used. It is there for conformance to IEEE488.2.

3.1.7 *STB?

Syntax

*STB?

Description

The *STB? command queries the instrument's *Status Byte Register*. This is a read-only register and the bits are not cleared when you read the register. See also figure [Status Structure](#) ^[65].

Table <%TABLENO++%>: *Status Byte Register*

Bit #	Decimal Value	Mnemonic - Description
0	1	AUX - Auxiliary Status Structure Summary Bit One or more bits are set in the <i>Auxiliary Event Register</i> (bits must be enabled, see Auxiliary Status Register Group ^[69]).
1	2	MEAS - Measurement Status Structure Summary Bit One or more bits are set in the <i>Measurement Event Register</i> (bits must be enabled, see Measurement Status Register Group ^[70]).
2	4	EAV - Error Available Bit One or more errors have been stored in the Error Queue. Use the SYSTEM:ERROR?[:NEXT] ^[16] command to read and delete errors.
3	8	QUES - Questionable Data Structure Summary Bit One or more bits are set in the <i>Questionable Event Register</i> (bits must be enabled, see Questionable Data Register Group ^[71]).
4	16	MAV - Message Available Bit Data is available in the instrument's output buffer.
5	32	ESB - Standard Event Status Bit One or more bits are set in the <i>Standard Event Register</i> (bits must be enabled, see *ESE ^[14] command).
6	64	RQS/MSS - Request Service / Master Summary Status Bit One or more bits are set in the <i>Status Byte Register</i> and may generate a Request for Service (RQS). Bits must be enabled using the *SRE ^[14] command.
7	128	OPER - Standard Operation Status Summary Bit One or more bits are set in the <i>Standard Operation Register</i> (bits must be enabled, see STATus:OPERation:ENABle ^[72] command).

3.1.8 *SRE, *SRE?

Syntax

*SRE <value>

*SRE?

Description

The *SRE command enables bits in the enable register (*Service Request Enable Register*) for the *Status Byte Register Group*. See table [Status Byte Register](#)^[13] for a description of the bits. Once enabled, the corresponding bits may generate a Request for Service (RQS) in the status byte. This RQS event may generate a "call back" to your application as a type of asynchronous interrupt.

3.1.9 *CLS

Syntax

*CLS

Description

The *CLS command clears the event registers in all register groups. This command also clears the error queue.

3.1.10 *ESR?, *ESE, *ESE?

Syntax

*ESR?

*ESE <value>

*ESE?

Description

The *ESR? command queries the *Standard Event Register*. In order to be reported to the *Status Byte Register*, the corresponding bits in the event register must be enabled using the *ESE command.

The *ESE command enables bits in the enable register (*Standard Event Status Enable Register*) for the *Standard Event Register Group*. The selected bits are then reported to bit 5 of the *Status Byte Register*.

See also chapter [Status Subsystem](#)^[65] and figure [Status Structure](#)^[65].

The following table lists the bit definitions for the *Standard Event Register Group*.

Table <%TABLENO++%>: *Standard Event Register*

Bit #	Decimal Value	Description
0	1	Operation Complete All commands prior to and including *OPC ^[12] have been executed
1	2	not used
2	4	Query Error The instrument tried to read the output buffer but it was empty or a new command line was received before a previous query has been read or both the input and output buffers are full.
3	8	Device Dependent Error A self-test or calibration error occurred (an error in the -300...-399 range or any positive error has been generated). For a complete listing of error messages, see chapter SCPI Error Messages ^[87] .

Bit #	Decimal Value	Description
4	16	Execution Error An execution error occurred (an error in the -200...-299 range has been generated).
5	32	Command Error A command syntax error occurred (an error in the -100...-199 range has been generated).
6	64	User Request
7	128	Power On Power has been turned off and on since the last time the event register was read or cleared.

Example

→ *ESE 4

Sets bit 2 of the *Standard Event Status Enable Register* which enables bit 2 of the *Standard Event Register* to be reported to bit 5 of the *Status Byte Register*.

3.2 System Subsystem Commands

Table <TABLENO++>: System Subsystem Commands

Command	Description
<code>SYSTem</code> ^[22]	
<code>:BEEPer</code> ^[16]	
<code>[:IMMediate]</code>	Issues an audible signal
<code>:STATE {ON 1 OFF 0}</code>	Activates/deactivates the beeper
<code>:STATe?</code>	Returns the state of the beeper
<code>:ERRor</code> ^[16]	
<code>[:NEXT]?</code>	Returns the latest <error code>, "Message"
<code>:LFRequency</code> ^[17] {50 60 AUTO}	Sets the line frequency to fixed 50/60Hz or auto detect
<code>:ACTual?</code>	Returns the used line frequency
<code>:LFRequency?</code> ^[17]	Returns the line frequency setting
<code>:VERSion?</code> ^[17]	Returns the level of SCPI standard (1999.0)

3.2.1 Beeper

Syntax

```
SYSTem:BEEPer[ :IMMediate ]
SYSTem:BEEPer:STATE {ON|1|OFF|0}
SYSTem:BEEPer:STATe?
```

Description

The `SYSTem:BEEPer[:IMMediate]` issues an audible signal.

The `SYSTem:BEEPer:STATe` command activates or deactivates the beeper.

Example

- `SYST:BEEP`
The instrument will issue an audible signal.
- `SYST:BEEP:STAT?`
Queries the state of the beeper activation.
- ← 1
Response for beeper activated. The response is ASCII 1 if the beeper is activated or an ASCII 0 if the beeper is deactivated.

3.2.2 Error Query

Syntax

```
SYSTem:ERRor[ :NEXT ]?
```

Description

The `SYSTem:ERRor[:NEXT]?` command queries the instrument's error queue. This is a query-only command. For a detailed list of possible error codes see chapter [SCPI Error Messages](#)^[81].

Example

→ SYST:ERR?
Query the error queue.

← +0,"No error"
Typical response.

3.2.3 Line Frequency

Syntax

```
SYSTem:LFRequency {50|60|AUTO}  
SYSTem:LFRequency:ACTual?  
SYSTem:LFRequency?
```

Description

The SYSTem:LFRequency sets the line frequency to use for signal filtering to the given value or to automatic detection.

With the SYSTem:LFRequency:ACTual? query the actual used value can be retrieved.

The SYSTem:LFRequency? query returns the line frequency setting.

Example

→ SYST:LFR 50
Sets the line frequency setting to 50Hz.

→ SYST:LFR:ACT?
Queries the actual used line frequency.

← 60
Typical response.

3.2.4 SCPI Version Query

Syntax

```
SYSTem:VERSion?
```

Description

The SYSTem:VERSion? command queries the level of the SCPI standard implemented. This is a query-only command.

Example

→ SYST:VERS?
Query the SCPI version.

← 1999.0
Typical response.

3.3 Measurement Commands

Table <TABLENO++>: Measurement Commands for a TED4000 Series Instrument

Command	Description
ABORT ^[22]	Aborts the current measurement
CONFigure[:SCALar] ^[22]	
[:TEMPerature]	Configures instrument for temperature measurement
:CURRent[1][:DC]	Configures instrument for TEC current measurement
:VOLTage[1][:DC]	Configures instrument for TEC voltage measurement
:POWer[1]	Configures instrument for TEC power measurement
:TSEnSor	Configures for sensor signal measurement
CONFigure? ^[22]	Returns the current configuration
INITiate[:IMMediate] ^[24]	Starts measurement
FETCh? ^[24]	Returns the last measurement data. The data returned is the data configured with the last CONFigure[:SCALar] or the last MEASure[:SCALar] command
FETCh ^[24]	
:TEMPerature?	Returns the last measured temperature
:CURRent[1][:DC]?	Returns the last measured TEC current
:VOLTage[1][:DC]?	Returns the last measured TEC voltage
:POWer[1]?	Returns the last measured TEC power
:TSEnSor?	Returns the last measured sensor signal
READ? ^[26]	Starts new measurement and returns data
MEASure[:SCALar] ^[26]	
[:TEMPerature]?	Measures temperature
:CURRent[1][:DC]?	Measures TEC current
:VOLTage[1][:DC]?	Measures TEC voltage
:POWer[1]?	Measures TEC power
:TSEnSor?	Measures sensor signal

Table <TABLENO++>: Measurement Commands for a LDC4000 Series Instrument

Command	Description
ABORt ^[22]	Aborts the current measurement
CONFIgure[:SCALar] ^[22]	
[:CURRent][1][:DC]	Configures instrument for LD current measurement
:VOLTage[1][:DC]	Configures instrument for LD voltage measurement
:CURRent2[:DC]	Configures instrument for PD current measurement
:POWer2	Configures instrument for power measurement via photodiode
:VOLTage2[:DC]	Configures instrument for thermopile/power meter voltage measurement
:POWer3	Configures instrument for power measurement via thermopile/power meter
:POWer[1]	Configures instrument for LD electrical input power measurement
CONFIgure? ^[22]	Returns the current configuration
INITiate[:IMMediate] ^[24]	Starts measurement
FETCh? ^[24]	Returns the last measurement data. The data returned is the data configured with the last CONFIgure[:SCALar] or the last MEASure[:SCALar] command
FETCh ^[24]	
:CURRent[1][:DC]?	Returns the last measured LD current
:VOLTage[1][:DC]?	Returns the last measured LD voltage
:CURRent2[:DC]?	Returns the last measured PD current
:POWer2?	Returns the last measured optical power (via PD)
:VOLTage2[:DC]?	Returns the last measured thermopile/power meter voltage
:POWer3?	Returns the last measured optical power (via thermopile/power meter)
:POWer[1]?	Returns the last measured LD input electrical power
READ? ^[26]	Starts new measurement and returns data
MEASure[:SCALar] ^[26]	
[:CURRent][1][:DC]?	Measures LD current
:VOLTage[1][:DC]?	Measures LD voltage
:CURRent2[:DC]?	Measures PD current
:POWer2?	Measures LD power via PD
:VOLTage2[:DC]?	Measures thermopile/power meter voltage
:POWer3?	Measures LD power via thermopile/power meter
:POWer[1]?	Measures LD electrical input power

Table <%TABLENO++%>: Measurement Commands for a ITC4000 Series Instrument

Command	Description
ABORT ^[22]	Aborts the current measurement
CONFigure[:SCALar] ^[22]	
:TEMPerature	Configures instrument for temperature measurement
:CURRent3[:DC]	Configures instrument for TEC current measurement
:VOLTage3[:DC]	Configures instrument for TEC voltage measurement
:POWer4	Configures instrument for TEC power measurement
:TSEnSor	Configures instrument for temperature sensor signal measurement
[:CURRent][1][:DC]	Configures instrument for LD current measurement
:VOLTage[1][:DC]	Configures instrument for LD voltage measurement
:CURRent2[:DC]	Configures instrument for PD current measurement
:POWer2	Configures instrument for for power measurement via photodiode
:VOLTage2[:DC]	Configures instrument for thermopile/power meter voltage measurement
:POWer3?	Configures instrument for power measurement via thermopile/power meter
:POWer[1]?	Configures instrument for LD electrical input power measurement
CONFigure? ^[22]	Returns the current configuration
INITiate[:IMMediate] ^[24]	Starts measurement
FETCh? ^[24]	Returns the last measurement data. The data returned is the data configured with the last CONFigure[:SCALar] or the last MEASure[:SCALar] command
FETCh ^[24]	
:TEMPerature?	Returns the last measured temperature
:CURRent3[:DC]?	Returns the last measured TEC current
:VOLTage3[:DC]?	Returns the last measured TEC voltage
:POWer4?	Returns the last measured TEC power
:TSEnSor?	Returns the last measured temperature sensor signal
:CURRent[1][:DC]?	Returns the last measured LD current
:VOLTage[1][:DC]?	Returns the last measured LD voltage
:CURRent2[:DC]?	Returns the last measured PD current
:POWer2?	Returns the last measured optical power (via PD)
:VOLTage2[:DC]?	Returns the last measured thermopile/power meter voltage
:POWer3?	Returns the last measured optical power (via thermopile/power meter)
:POWer[1]?	Returns the last measured LD input electrical power
READ? ^[26]	Starts new measurement and returns data
MEASure[:SCALar] ^[26]	
:TEMPerature?	Measures temperature
:CURRent3[:DC]?	Measures TEC current
:VOLTage3[:DC]?	Measures TEC voltage
:POWer4?	Measures TEC power

Command	Description
:TSENsOr?	Measures temperature sensor signal
:CURRent[1][:DC]?	Measures LD current
:VOLTage[1][:DC]?	Measures LD voltage
:CURRent2[:DC]?	Measures PD current
:POWER2?	Measures LD power via PD
:VOLTage2[:DC]?	Measures thermopile/power meter voltage
:POWER3?	Measures LD power via thermopile/power meter
:POWER[1]?	Measures LD electrical input power

3.3.1 Abort Measurement in Progress

Syntax

ABORt

Description

This command aborts a measurement in progress.

Remarks

This command aborts a measurement a measurement in progress and stops, returning the instrument to the idle state.

Example

→ ABOR
Abort the measurement in progress.

3.3.2 Configuring a Measurement

Syntax

For TED4000 Series instruments:

```
CONFigure[:SCALar][:TEMPerature]  
CONFigure[:SCALar]:CURRENT[1][:DC]  
CONFigure[:SCALar]:VOLTage[1][:DC]  
CONFigure[:SCALar]:POWER[1]  
CONFigure[:SCALar]:TSENSOR  
CONFigure?
```

For LDC4000 Series instruments:

```
CONFigure[:SCALar][:CURRENT][1][:DC]  
CONFigure[:SCALar]:VOLTage[1][:DC]  
CONFigure[:SCALar]:CURRENT2[:DC]  
CONFigure[:SCALar]:POWER2  
CONFigure[:SCALar]:VOLTage2[:DC]  
CONFigure[:SCALar]:POWER3  
CONFigure[:SCALar]:POWER[1]  
CONFigure?
```

For ITC4000 Series instruments:

```
CONFigure[:SCALar]:TEMPerature  
CONFigure[:SCALar]:CURRENT3[:DC]  
CONFigure[:SCALar]:VOLTage3[:DC]  
CONFigure[:SCALar]:POWER4  
CONFigure[:SCALar]:TSENSOR  
CONFigure[:SCALar][:CURRENT][1][:DC]  
CONFigure[:SCALar]:VOLTage[1][:DC]  
CONFigure[:SCALar]:CURRENT2[:DC]  
CONFigure[:SCALar]:POWER2  
CONFigure[:SCALar]:VOLTage2[:DC]  
CONFigure[:SCALar]:POWER3  
CONFigure[:SCALar]:POWER[1]  
CONFigure?
```


Description

Configures the instrument to measure the selected value. The `CONFigure` command does not perform the measurement. Use the `INITiate` or `READ?` command to take the configured measurement. The configuration query returns the current measurement configuration.

Example

The following program segment configures a TED4000 Series instrument for temperature measurements and performs one measurement with the `READ?` command.

```
→ CONF:TEMP
   Configure the instrument for temperature measurement.
→ READ?
   Perform a measurement and send the reading to the output buffer.
← 2.500000E+01
   Typical response.
```

Example

The following program segment configures a TED4000 Series instrument for temperature measurements and performs one measurement with the `INITiate[:IMMEDIATE]` and `FETCh?` command sequence.

```
→ CONF:TEMP
   Configure the instrument for temperature measurement.
→ INIT
   Perform the measurement and store reading in memory.
→ FETC?
   Transfer reading from memory to instrument's output buffer.
← 2.500000E+01
   Typical response.
```

Example

```
→ CONF?
   Query the current measurement configuration.
← TEMP
   Typical response.
```

Example

The following program segment configures a LDC4000 Series instrument for thermopile/power meter voltage measurements and performs one measurement with the `INITiate[:IMMEDIATE]` and `FETCh?` command sequence.

```
→ CONF:VOLT2
   Configure the instrument for thermopile/power meter voltage measurement.
→ INIT
   Perform the measurement and store reading in memory.
→ FETC?
   Transfer reading from memory to instrument's output buffer.
```

← 3.770000E-01
Typical response.

Example

The following program segment configures a ITC4000 Series instrument for TEC current measurement and performs one measurement with the `READ?` command.

→ `CONF:CURR3`
Configure the instrument for TEC current measurement.

→ `READ?`
Perform a measurement and send the reading to the output buffer.

← 6.532000E+00
Typical response.

3.3.3 Initiating a Measurement

Syntax

`INITiate[:IMMediate]`

Description

This command performs a previously configured measurement, and stores the reading in memory.

Example

The following program segment configures a TED4000 / ITC4000 Series Instrument for temperature measurements and performs one measurement.

→ `CONF:TEMP`
Configure the instrument for temperature measurement.

→ `INIT`
Perform the measurement and store reading in memory.

→ `FETC?`
Transfer reading from memory to instrument's output buffer.

← 2.500000E+01
Typical response.

3.3.4 Fetch a Measurement

Syntax

For TED4000 Series instruments:

`FETCh?`
`FETCh:TEMPerature?`
`FETCh:CURREnt[1][:DC]?`
`FETCh:VOLTage[1][:DC]?`
`FETCh:POWER[1]?`
`FETCh:TSENSor?`

For LDC4000 Series instruments:

`FETCh?`

```

FETCh:CURRent[1][:DC]?
FETCh:VOLTage[1][:DC]?
FETCh:CURRent2[:DC]?
FETCh:POWer2?
FETCh:VOLTage2[:DC]?
FETCh:POWer3?
FETCh:POWer[1]?

```

For ITC4000 Series instruments:

```

FETCh?
FETCh:TEMPerature?
FETCh:CURRent3[:DC]?
FETCh:VOLTage3[:DC]?
FETCh:POWer4?
FETCh:TSENSor?
FETCh:CURRent[1][:DC]?
FETCh:VOLTage[1][:DC]?
FETCh:CURRent2[:DC]?
FETCh:POWer2?
FETCh:VOLTage2[:DC]?
FETCh:POWer3?
FETCh:POWer[1]?

```

Description

This command transfers readings from memory to the instrument's output buffer where you can read them into your computer. The readings are not erased from memory when you read them. You can send the command multiple times to retrieve the same data. With the `FETCh?` command the last configured measurement will be transferred to the instruments output buffer. With `FETCh:XXX?` the value of the last measurement of XXX will be transferred to the instruments output buffer.

Example

```

→ FETC?
   Query last measurement.

← 2.500000E+01
   Typical response.

```

Example

```

→ FETC:POW2?
   Query last measurement of optical power via photodiode of an ITC4000 Series
   instrument.

← 4.520000E-03
   Typical response.

```

3.3.5 Read Measurement Data

Syntax

READ?

Description

The `READ?` command performs a previously configured measurement, and then sends the reading to the instrument output buffer. It has the same effect as `INITiate[:IMMEDIATE]` followed by the `FETCh?` command.

Example

The following program segment configures a TED4000 / ITC4000 Series instrument for temperature measurements and performs one measurement.

```
→ CONF:TEMP
   Configure the instrument for temperature measurement.

→ READ?
   Perform a measurement and send the reading to the output buffer.

← 2.500000E+01
   Typical response.
```

3.3.6 Simple Measurement

Syntax

For TED4000 Series instruments:

```
MEASure[:SCALar][:TEMPerature]?
MEASure[:SCALar][:CURRent][1][:DC]?
MEASure[:SCALar]:VOLTage[1][:DC]?
MEASure[:SCALar]:POWER[1]?
MEASure[:SCALar]:TSENSor?
```

For LDC4000 Series instruments:

```
MEASure[:SCALar][:CURRent][1][:DC]?
MEASure[:SCALar]:VOLTage[1][:DC]?
MEASure[:SCALar]:CURRent2[:DC]?
MEASure[:SCALar]:POWER2?
MEASure[:SCALar]:VOLTage2[:DC]?
MEASure[:SCALar]:POWER3?
MEASure[:SCALar]:POWER[1]?
```

For ITC4000 Series instruments:

```
MEASure[:SCALar]:TEMPerature?
MEASure[:SCALar]:CURRent3[:DC]?
MEASure[:SCALar]:VOLTage3[:DC]?
MEASure[:SCALar]:POWER4?
MEASure[:SCALar]:TSENSor?
MEASure[:SCALar][:CURRent][1][:DC]?
MEASure[:SCALar]:VOLTage[1][:DC]?
MEASure[:SCALar]:CURRent2[:DC]?
MEASure[:SCALar]:POWER2?
```

MEASure[:SCALar]:VOLTage2[:DC]?

MEASure[:SCALar]:POWER3?

MEASure[:SCALar]:POWER[1]?

Description

The `MEASure?` command provides the easiest way to program the instrument for measurements. However, this command does not offer much flexibility. The results are sent directly to the instrument's output buffer. Sending `MEASure?` is functionally the same as sending `CONFigure` followed immediately by a `READ?` command.

Example

→ `MEAS:TEMP?`

Query temperature of a TED4000 / ITC4000 Series instrument.

← `2.500000E+01`

Typical response.

3.4 LD Output Commands

The `OUTPut[1]` subsystem controls the LD output of the instrument.

Note

These commands are valid for LDC4000 and ITC4000 Series instruments only. For LDC4000 and ITC4000 Series instruments the command suffix is 1 (can be omitted).

Table <%TABLENO++%>: LD Output Commands

Command	Description
<code>OUTPut[1]</code>	Path to LD output
<code>[:STATe]^[29] { OFF ON 0 1 }</code>	Set output state
<code>:DELay^[29] { MIN MAX DEF <seconds> }</code>	Set switch-on delay
<code>:POLarity^[30] { NORMal CG INVerted AG }</code>	Set output polarity
<code>:FILTer^[30][:LPASS]</code>	Path to LD output low pass filter
<code>[:STATe] { ON 1 OFF 0 }</code>	Enables/disables LD output low pass filter (since firmware V1.5)
<code>[:STATe] ?</code>	Returns output filter state (since firmware V1.5)
<code>:PROTection</code>	
<code>:VOLTagE</code>	
<code>[:LEVel]^[31] { MIN MAX DEF <volts> }</code>	Set output protection voltage
<code>[:LEVel] ?^[31] { MIN MAX DEF }</code>	Return output protection voltage
<code>:TRIPped?^[33]</code>	Return voltage protection tripped
<code>:EXTeRnal</code>	
<code>[:MODE]^[32] { OFF PROTection ENABle }</code>	Set external protection mode
<code>:TRIPped?^[33]</code>	Return external protection tripped
<code>:INTeRnal</code>	
<code>[:MODE]^[31] { OFF PROTection ENABle }</code>	Set internal protection mode
<code>:TRIPped?^[33]</code>	Return internal protection tripped
<code>:INTLock</code>	
<code>:TRIPped?^[33]</code>	Return interlock protection tripped
<code>:KEYLock</code>	
<code>:TRIPped?^[33]</code>	Return front panel key switch protection tripped
<code>:OTEMp</code>	
<code>:TRIPped?^[33]</code>	Return over temperature protection tripped

3.4.1 Output State

Syntax

```
OUTPut[1][:STATe] {OFF|0|ON|1}
```

```
OUTPut[1][:STATe]?
```

Description

The `OUTPut[1][:STATe]` command switches the instrument's LD output on/off.

Example

```
→ OUTP ON
```

Switches the LD output on.

```
→ OUTP?
```

Queries the state of the LD output.

```
← 1
```

The response is ASCII 1 if the output is switched on or an ASCII 0 if the output is off.

Note

This command is valid for LDC4000 and ITC4000 Series instruments only.

Note

The `*RST` command switches the output off.

3.4.2 Switch-On Delay

Syntax

```
OUTPut[1]:DElay {MIN|MAX|DEF|<seconds>}
```

```
OUTPut[1]:DElay? [{MIN|MAX|DEF}]
```

Description

The `OUTPut[1]:DElay` sets the output switch-on delay of the LD output.

Example

```
→ OUTP:DEL 3
```

Sets the LD output switch-on delay to 3 seconds.

Note

This command is valid for LDC4000 and ITC4000 Series instruments only.

3.4.3 Polarity

Syntax

```
OUTPut[1]:POLarity {NORMal|CG|INVerted|AG}  
OUTPut[1]:POLarity?
```

Description

The `OUTPut[1]:POLarity` command sets the LD output polarity to cathode ground (`NORMal` or `CG`) or anode ground (`INVerted` or `AG`). The parameters `NORMal` and `INVerted` are for SCPI conformance.

Example

- `OUTP:POL CG`
Sets the LD output polarity to cathode ground.
- `OUTP:POL?`
Queries the LD output polarity.
- ← `CG`
The response is either `CG` or `AG`.

Note

This command is valid for LDC4000 and ITC4000 Series instruments only.

3.4.4 Noise Reduction Filter

Syntax

```
:FILTer[:LPASs]:STATe {ON|1|OFF|0}  
:FILTer[:LPASs]:STATe?
```

Description

The `FILTer[:LPASs]:STATe` command switches the LD output low pass filter on or off.

Example

- `FILT:STAT ON`
Switches LD output noise reduction filter ON.
- `FILT:STAT?`
Queries the LD output noise filter status.
- ← The response is ASCII 1 if the filter is switched on or an ASCII 0 if the output is off.

Note

This command is valid for LDC4005, ITC4001, ITC4002QCL, ITC4005 and ITC4005QCL instruments, FW version 1.5 up, only.

3.4.5 Protection Voltage

Syntax

```
OUTPut[1]:PROTection:VOLTagE[:LEVel] {MIN|MAX|DEF|<volts>}
OUTPut[1]:PROTection:VOLTagE[:LEVel]? [{MIN|MAX|DEF}]
```

Description

The `OUTPut[1]:PROTection:VOLTagE[:LEVel]` command sets the output voltage protection (compliance voltage) threshold. If this threshold is reached the output switches off.

Example

```
→ OUTP:PROT:VOLT 8.0
   Sets the LD output protection voltage to 8V.

→ OUTP:PROT:VOLT? MAX
   Queries the maximum settable LD output protection voltage.

← 1.000000E+01
   Typical response.
```

Note

This command is valid for LDC4000 and ITC4000 Series instruments only.

3.4.6 Temperature Protection Mode

Syntax

```
OUTPut[1]:PROTection:INTernal[:MODE] {OFF|PROTection|ENABle}
OUTPut[1]:PROTection:INTernal[:MODE]?
```

Description

The `OUTPut[1]:PROTection:INTernal[:MODE]` command sets the function of the internal temperature protection circuit.

Parameter	Description
OFF	Temperature protection is disabled and has no effect on the LD output
PROTection	Acts as protection. If the temperature is out of bounds the LD output is switched off
ENABle	Acts as enable. If the temperature is out of bounds the LD output is disabled temporarily.

Example

```
→ OUTP:PROT:INT ENAB
   Sets the temperature protection mode to output-enable mode.
```

Note

This command is valid for ITC4000 Series instruments only.

See also [Temperature Protection](#)^[62] for further details.

3.4.7 LD-ENABLE Input Mode

Syntax

```
OUTPut[1]:PROTection:EXTeRnal[:MODE] {OFF|PROTection|ENABle}
```

```
OUTPut[1]:PROTection:EXTeRnal[:MODE]?
```

Description

The `OUTPut[1]:PROTection:EXTeRnal[:MODE]` command sets the function of the rear panel LD-ENABLE input.

Parameter	Description
OFF	The LD-ENABLE input is ignored and has no effect
PROTection	Acts as protection input (inhibit functionality). A TTL low level switches the LD output off.
ENABle	Acts as enable input. A TTL low level disables the LD output temporarily.

Example

```
→ OUTP:PROT:EXT ENAB
```

Sets the LD-ENABLE input mode to output-enable mode.

Note

This command is valid for LDC4000 and ITC4000 Series instruments only.

3.4.8 Protection Queries

Syntax

```
OUTPut[1]:PROTection:VOLTagE:TRIPped?  
OUTPut[1]:PROTection:EXTeRnal:TRIPped?  
OUTPut[1]:PROTection:INTeRnal:TRIPped?  
OUTPut[1]:PROTection:INTLock:TRIPped?  
OUTPut[1]:PROTection:KEYLock:TRIPped?  
OUTPut[1]:PROTection:OTEMp:TRIPped?
```

Description

The `OUTPut[1]:PROTection:VOLTagE:TRIPped?` query command returns 1 if the LD output protection voltage was reached.

The `OUTPut[1]:PROTection:EXTeRnal:TRIPped?` query command returns 1 if the LD-ENABLE input has a TTL low level.

The `OUTPut[1]:PROTection:INTeRnal:TRIPped?` query command returns 1 if the temperature window protection is currently active. This command is valid for ITC4000 instruments only.

The `OUTPut[1]:PROTection:INTLock:TRIPped?` query command returns 1 if the interlock circuit is open.

The `OUTPut[1]:PROTection:KEYLock:TRIPped?` query command returns 1 if the front panel key switch is in the lock position.

The `OUTPut[1]:PROTection:OTEMp:TRIPped?` query command returns 1 if the instrument's internal temperature is too high.

Example

```
→ OUTP:PROT:KEYL:TRIP?  
   Queries the front panel key switch.  
  
← 0  
   Typical response.
```

Note

These commands are valid for LDC4000 and ITC4000 only.

3.5 PD Input Commands

The `INPut[1]` subsystem controls the PD input of the instrument.

Note

These commands are valid for LDC4000 and ITC4000 Series instruments only. For LDC4000 and ITC4000 Series instruments the command suffix is 1 (can be omitted).

Table <%TABLENO++%>: PD Input Commands

Command	Description
<code>INPut[1]</code>	Path to PD input
<code>:POLarity^[34] {NORMAL CG INVERTed AG}</code>	Set input state
<code>:POLarity?^[34]</code>	Return input state
<code>:BIAS[:STATE]^[35] {OFF 0 ON 1}</code>	Switch PD-Bias on or off
<code>:BIAS[:STATE]?^[35]</code>	Return PD-Bias state
<code>:BIAS:VOLTage[:DC]^[35] {MIN MAX DEF <volts>}</code>	Set PD-Bias voltage level
<code>:BIAS:VOLTage[:DC]?^[35] [{MIN MAX DEF}]</code>	Return PD-Bias voltage
<code>:ROUTE[:TERMinals]^[36] {DSUB BNC}</code>	Set PD-input routing
<code>:ROUTE[:TERMinals]?^[36]</code>	Return PD-input routing

See also chapter [PD Sense Commands^{\[45\]}](#).

3.5.1 Polarity

Syntax

```
INPut[1]:POLarity {NORMAL|CG|INVERTed|AG}
INPut[1]:POLarity?
```

Description

The `INPut[1]:POLarity` command sets the PD input polarity to cathode ground (`NORMAL` or `CG`) or anode ground (`INVERTed` or `AG`). The parameters `NORMAL` and `INVERTed` are for SCPI conformance.

Example

- `INP:POL CG`
Sets the PD input polarity to cathode ground.
- `INP:POL?`
Queries the PD input polarity.
- ← `CG`
The response is either `CG` or `AG`.

Note

This command is valid for LDC4000 and ITC4000 Series instruments only.

3.5.2 BIAS State

Syntax

```
INPut[1]:BIAS[:STATe] {OFF|0|ON|1}
INPut[1]:BIAS[:STATe]?
```

Description

The `INPut[1]:BIAS[:STATe]` command switches the instrument's PD input BIAS on/off.

Example

- `INP:BIAS ON`
Switches the PD input BIAS on.
- `INP:BIAS?`
Queries the state of the PD input BIAS.
- ← `1`
The response is ASCII `1` if the BIAS is switched on or an ASCII `0` if the BIAS is off.

Note

This command is valid for LDC4000 and ITC4000 Series instruments only.

Note

The `*RST11` command switches the BIAS off.

3.5.3 BIAS Voltage

Syntax

```
INPut[1]:BIAS:VOLTage[:DC] {MIN|MAX|DEF|<volts>}
INPut[1]:BIAS:VOLTage[:DC]? [{MIN|MAX|DEF}]
```

Description

The `INPut[1]:BIAS:VOLTage[:DC]` command sets the PD input BIAS voltage.

Example

- `INP:BIAS:VOLT 800mV`
Sets the PD input BIAS voltage to 0.8V.
- `INP:BIAS:VOLT? MAX`
Queries the maximum settable PD input BIAS voltage.
- ← `5.000000E+01`
Typical response.

Note

This command is valid for LDC4000 and ITC4000 Series instruments only.

3.5.4 Input Routing

Syntax

```
INPut[1]:ROUTe[:TERMinals] {DSUB|BNC}  
INPut[1]:ROUTe[:TERMinals]?
```

Description

The `INPut[1]:ROUTe[:TERMinals]` command sets the path for the PD input routing to either the rear panel LASER OUTPUT D-SUB connector or the rear panel BNC connector.

Example

- `INP:ROUT BNC`
Sets the path for the PD input to the rear panel BNC connector.
- `INP:ROUT?`
Queries the path for the PD input routing.
- ← `BNC`
The answer is either `DSUB` or `BNC`.

Note

This command is valid for LDC4000 and ITC4000 Series instruments only.

3.6 Thermopile/Power Meter Input Commands

The `INPut2` subsystem controls the thermopile/power meter input of the instrument.

Note

These commands are valid for LDC4000 and ITC4000 Series instruments only.

Table <%TABLENO++%>: Thermopile/Power Meter Input Commands

Command	Description
<code>INPut[2]</code>	Path to PD input
<code>:ROUTE[:TERMinals] {DSUB BNC}</code>	Set thermopile/power meter input routing
<code>:ROUTE[:TERMinals]?</code>	Return thermopile/power meter input routing

See also chapter [Thermopile/Power Meter Sense Commands](#).

3.6.1 Input Routing

Syntax

```
INPut2:ROUTE[:TERMinals] {DSUB|BNC}
```

```
INPut2:ROUTE[:TERMinals]?
```

Description

The `INPut2:ROUTE[:TERMinals]` command sets the path for the thermopile/power meter input routing either to the rear panel LASER OUTPUT D-SUB connector or to the rear panel BNC connector.

Example

- `INP2:ROUT BNC`
Sets the path for the thermopile/power meter input to the rear panel BNC connector.
- `INP2:ROUT?`
Queries the path for the thermopile/power meter input routing.
- ← `BNC`
The answer is either `DSUB` or `BNC`.

Note

This command is valid for LDC4000 and ITC4000 Series instruments only.

3.7 LD Source Commands

The `SOURCE[1]` subsystem controls the LD source part of the instrument.

Note

These commands are valid for LDC4000 and ITC4000 Series instruments only.

Table <%TABLENO++%>: LD Source Commands

Command	Description
<code>SOURCE[1]</code>	Path to LD source function
<code>:FUNCTION</code> ^[40]	
<code>[:MODE]</code> ^[53] { <code>CURRENT</code> <code>POWER</code> }	Set LD driver source function
<code>[:MODE]</code> ^{[53]?}	Return LD driver source function
<code>[:SHAPE]</code> { <code>DC</code> <code>PULSE</code> }	Set CW(DC) or QCW(PULSE) mode
<code>[:SHAPE]?</code>	Return mode
<code>:CURRENT</code>	
<code>:LIMIT</code>	
<code>[:AMPLITUDE]</code> { <code>MIN</code> <code>MAX</code> <code><amps></code> } ^[40]	Set LD limit current value
<code>[:AMPLITUDE]?</code> [{ <code>MIN</code> <code>MAX</code> }] ^[40]	Return LD limit current value
<code>:TRIPPED?</code> ^[40]	Return LD limit current tripped
<code>[:LEVEL] [:IMMEDIATE] [:AMPLITUDE]</code> ^[41] { <code>MIN</code> <code>MAX</code> <code><amps></code> }	Set LD current setpoint
<code>[:LEVEL] [:IMMEDIATE] [:AMPLITUDE]?</code> ^[40] [{ <code>MIN</code> <code>MAX</code> }]	Return LD current setpoint
<code>:POWER</code>	
<code>[:LEVEL]</code> ^[41]	
<code>[:IMMEDIATE] [:AMPLITUDE]</code> { <code>MIN</code> <code>MAX</code> <code><watts></code> }	Set LD power setpoint
<code>[:IMMEDIATE] [:AMPLITUDE]?</code> [{ <code>MIN</code> <code>MAX</code> }]	Return LD power setpoint
<code>:DIODE [:CURRENT] [:IMMEDIATE]</code> [:AMPLITUDE] { <code>MIN</code> <code>MAX</code> <code><amps></code> }	Set LD power via setting photodiode current setpoint
<code>:DIODE [:CURRENT] [:IMMEDIATE]</code> [:AMPLITUDE]? [{ <code>MIN</code> <code>MAX</code> }]	Return photodiode current setpoint
<code>:PMETER [:VOLTAGE] [:IMMEDIATE]</code> [:AMPLITUDE] { <code>MIN</code> <code>MAX</code> <code><volts></code> }	Set LD power via setting thermopile/power meter voltage setpoint
<code>:PMETER [:VOLTAGE] [:IMMEDIATE]</code> [:AMPLITUDE]? [{ <code>MIN</code> <code>MAX</code> }]	Return thermopile/power meter voltage setpoint
<code>:ALC</code> ^[43]	
<code>:SOURCE</code> { <code>DIODE</code> <code>PMETER</code> <code>PDIODE</code> <code>THERMOPILE</code> }	Set the power source feedback loop input

Command	Description
:SOURce?	Return the power source feedback loop input
:SPEed {MIN MAX DEF <percent>}	Set feedback loop speed setting
:SPEed? [{MIN MAX DEF}]	Return feedback loop speed setting
:AM ⁴³	
[:STATe] {OFF 0 ON 1}	Enables (ON) or disables (OFF) modulation
[:STATe]?	Return modulation state
:SOURce {INTernal EXTernal} [, {INTernal EXTernal}]	Set modulation source
:SOURce?	Return modulation source
:INTernal	
:SHApe {SINusoid SQUare TRIangle}	Set internal modulation shape
:SHApe?	Return internal modulation shape
:FREQuency {MIN MAX DEF <hertz>}	Set internal modulation frequency
:FREQuency? [{MIN MAX DEF}]	Return internal modulation frequency
[:DEPTh] {MIN MAX DEF <percent>}	Set internal modulation depth
[:DEPTh]? [{MIN MAX DEF}]	Return internal modulation depth
:PULSe ⁴⁴	
:PERiod {MIN MAX DEF <secs>}	Set pulse period
:PERiod? [{MIN MAX DEF}]	Return pulse period
:WIDTh {MIN MAX DEF <secs>}	Set pulse width
:WIDTh? [{MIN MAX DEFault}]	Return pulse width
:DCYCLE {MIN MAX DEF <percent>}	Set pulse duty cycle
:DCYCLE? [{MIN MAX DEF}]	Return pulse duty cycle
:HOLD {WIDTh DCYCLE}	Set parameter to hold
:HOLD?	Return parameter to hold

3.7.1 LD Source Function

Syntax

```
SOURce[1]:FUNction:MODE {CURRent|POWER}
SOURce[1]:FUNction:MODE?
SOURce[1]:FUNction[:SHAPE] {DC|PULSe}
SOURce[1]:FUNction[:SHAPE]?
```

Description

The `SOURce[1]:FUNction:MODE` and `SOURce[1]:FUNction[:SHAPE]` commands set the instrument's LD source function to constant current (CW) mode or to pulsed constant current (QCW) mode or to constant power (CW) mode.

Note

The combination `POWER` and `PULSe` is not available.

Example

- `SOUR:FUNC:MODE CURR;SHAP DC`
Sets the LD source function of a LDC4000 Series or an ITC4000 Series instrument to constant current (CW) mode.
- `SOUR:FUNC:MODE?;SHAP?`
Queries the LD source function.
- ← `CURR;DC`
Response for constant current (CW) mode or
- ← `CURR;PULS`
response for pulsed constant current (QCW) mode or
- ← `POW;DC`
response for constant power (CW) mode.

Note

This command is valid for LDC4000 and ITC4000 Series instruments only.

3.7.2 LD Current Limit

Syntax

```
SOURce[1]:CURRent:LIMit[:AMPLitude] {MIN|MAX|<amps>}
SOURce[1]:CURRent:LIMit[:AMPLitude]? [{MIN|MAX}]
SOURce[1]:CURRent:LIMit:TRIPped?
```

Description

The `SOURce[1]:CURRent:LIMit[:AMPLitude]` command sets the LD source limit current in amperes.

The `SOURce[1]:CURRent:LIMit:TRIPped?` query command returns an ASCII 1 if the LD source current reached the limit value or an ASCII 0 if it's inactive.

Example

- `SOUR:CURR:LIM 1.5`
Sets the LD source limit current to 1.5 A on a LDC4000 Series or an ITC4000 Series instrument.

- SOUR:CURR:LIM? MAX
Queries the maximum settable LD source limit current of a LDC4000 Series or an ITC4000 Series instrument.
- ← 1.500000E+01
Typical response.

Note

This command is valid for LDC4000 and ITC4000 Series instruments only.

3.7.3 LD Current Setpoint

Syntax

```
SOURce[1]:CURRent[:LEVel][:IMMediate][:AMPLitude] {MIN|MAX|<amps>}
SOURce[1]:CURRent[:LEVel][:IMMediate][:AMPLitude]? [{MIN|MAX}]
```

Description

The SOURce[1]:CURRent[:LEVel][:IMMediate][:AMPLitude] command sets the LD current setpoint in amperes in current source modes.

Example

- SOUR:CURR 1.0
Sets the LD source current to 1.0 A on a LDC4000 Series or an ITC4000 Series instrument.
- SOUR:CURR? MAX
Queries the maximum settable LD source current setpoint of a LDC4000 Series or an ITC4000 Series instrument.
- ← 2.000000E+01
Typical response.

Note

This command is valid for LDC4000 and ITC4000 Series instruments only.

3.7.4 LD Optical Power

Syntax

```
SOURce[1]:POWer[:LEVel][:IMMediate][:AMPLitude] {MIN|MAX|<watts>}
SOURce[1]:POWer[:LEVel][:IMMediate][:AMPLitude]? [{MIN|MAX}]
SOURce[1]:POWer:ALC:SOURce {DIODE|PMETER}
SOURce[1]:POWer:ALC:SOURce?
SOURce[1]:POWer:ALC:BANDwidth {MIN|MAX|DEF|<hertz>}
SOURce[1]:POWer:ALC:BANDwidth? [{MIN|MAX|DEF}]
```

```
SOURce[1]:POWer[:LEVel]:DIODE[:CURRent][:IMMediate]
  [:AMPLitude] {MIN|MAX|<amps>}
SOURce[1]:POWer[:LEVel]:DIODE[:CURRent][:IMMediate]
  [:AMPLitude]? [{MIN|MAX}]
SOURce[1]:POWer[:LEVel]:PMETER[:VOLTage][:IMMediate]
  [:AMPLitude] {MIN|MAX|<volts>}
SOURce[1]:POWer[:LEVel]:PMETER[:VOLTage][:IMMediate]
  [:AMPLitude]? [{MIN|MAX}]
```

Description

The `SOURce[1]:POWer[:LEVel][:IMMediate][:AMPLitude]` command sets the LD power setpoint in watts for constant power (CW) mode.

The `SOURce[1]:POWer:ALC:SOURce` command sets the feedback source loop input to current input (`DIODE`) or to voltage input (`PMETer`).

The `SOURce[1]:POWer:ALC:BANDwidth` command sets the bandwidth of the feedback source loop input.

For constant power (CW) mode the feedback source input - photodiode current or thermopile voltage - can additionally be used as parameter to control the LD output power.

The `SOURce[1]:POWer[:LEVel]:DIODE[:CURRent][:IMMediate][:AMPLitude]` command sets the value for the photodiode current feedback. The LD power will be regulated so that the current through the photodiode caused by the laser light will equal the value set by this command.

The `SOURce[1]:POWer[:LEVel]:PMETer[:VOLTage][:IMMediate][:AMPLitude]` command sets the value for the thermopile voltage feedback. The LD power will be regulated so that the voltage of the thermopile caused by the laser light will equal the value set by this command.

Example

- `SOUR:POW 0.001`
Sets the LD source power to 1.0 mW on a LDC4000 Series or an ITC4000 Series instrument.
- `SOUR:POW:ALC:SOUR PMET`
Sets the feedback source loop input to thermopile/power meter input.
- `SOUR:POW? MAX`
Queries the maximum settable LD source power setpoint of a LDC4000 Series or an ITC4000 Series instrument.
- ← `5.111964E-01`
Typical response.

Note

The power regulation is actually based on photodiode current or the thermopile/power meter voltage. Setting a new value for the responsivity for the connected photodiode or the connected thermopile/power meter will not cause the instrument to change its output setting. In this case the power set point will be adapted.

Example: The LD power is set to 1.0 mW, the photodiode responsivity is set to 1.0 A/W, the resulting feedback current of the photodiode is 1.0 mA. When you change the responsivity to 2.0 A/W the feedback current remains constant at 1.0 mA but the read back value of the laser power set point will decrease to 0.5 mW.

Note

This command is valid for LDC4000 and ITC4000 Series instruments only.

3.7.5 LD Power Feedback

Syntax

```
SOURce[1]:POWER:ALC:SOURce {DIODE|PMETER|PDIODE|THERMOPILE}
SOURce[1]:POWER:ALC:SOURce?
SOURce[1]:POWER:ALC:SPEED {MIN|MAX|DEF|<percent>}
SOURce[1]:POWER:ALC:SPEED? [ {MIN|MAX|DEF} ]
```

Description

The `SOURce[1]:POWER:ALC:SOURce` command sets the power source feedback loop input to photodiode (`DIODE|PDIODE`) or to thermopile/power meter (`PMETER|THERMOPILE`).

The `SOURce[1]:POWER:ALC:SPEED` command sets the speed of the power source feedback loop. Please refer to the instrument's operation manual for a detailed description of the constant power mode operation.

Example

- `SOUR:POW:ALC:SOUR PDI`
Sets the power source feedback loop input to photodiode.
- `SOUR:POW:ALC:BAND 250`
Sets the power source feedback loop bandwidth to 250 Hz.

Note

This command is valid for LDC4000 and ITC4000 Series instruments only.

3.7.6 LD Amplitude Modulation

Syntax

```
SOURce[1]:AM[:STATE] {OFF|0|ON|1}
SOURce[1]:AM[:STATE]?
SOURce[1]:AM:SOURce {INTERNAL|EXTERNAL}[,{INTERNAL|EXTERNAL}]
SOURce[1]:AM:SOURce?
SOURce[1]:AM:INTERNAL:SHAPE {SINUSOID|SQUARE|TRIANGLE}
SOURce[1]:AM:INTERNAL:SHAPE?
SOURce[1]:AM:INTERNAL:FREQUENCY {MIN|MAX|DEF|<hertz>}
SOURce[1]:AM:INTERNAL:FREQUENCY? [ {MIN|MAX|DEF} ]
SOURce[1]:AM:INTERNAL[:DEPTH] {MIN|MAX|DEF|<percent>}
SOURce[1]:AM:INTERNAL[:DEPTH]? [ {MIN|MAX|DEF} ]
```

Description

The `SOURce[1]:AM[:STATE]` command enables (`ON|1`) or disables (`OFF|0`) the amplitude modulation of the LD source.

The `SOURce[1]:AM:SOURce` command selects the amplitude modulation source(s) of the LD source. It may specify a single source or two sources. The specified sources, in the `SOURce[1]:AM:SOURce` command, are all selected and turned on. Any sources from a previous selection that are not part of the current selection list are deselected and turned off.

The `SOURce[1]:AM:INTERNAL:SHAPE` command sets the internal modulation shape.

The `SOURce[1]:AM:INTERNAL` command sets the internal modulation depth.

Example

- SOUR:AM 1
Enables LD source modulation on a LDC4000 Series or an ITC4000 Series instrument.
- SOUR:AM:SOUR EXT
Selects the external BNC connector as source for modulation.
- SOUR:AM:SOUR INT,EXT
Selects both, internal modulator and external BNC connector as source for modulation.
- SOUR:AM:INT:FREQ? DEF
Queries the default internal modulation frequency.
- ← 1.000000E+03
Typical response.

Note

This command is valid for LDC4000 and ITC4000 Series instruments only.

3.7.7 QCW Pulse Setting

Syntax

SOURce[1]:PULSe:PERiod {MIN|MAX|DEF|<secs>}
SOURce[1]:PULSe:PERiod? [{MIN|MAX|DEF}]
SOURce[1]:PULSe:WIDTh {MIN|MAX|DEF|<secs>}
SOURce[1]:PULSe:WIDTh? [{MIN|MAX|DEFault}]
SOURce[1]:PULSe:DCYClE {MIN|MAX|DEF|<percent>}
SOURce[1]:PULSe:DCYClE? [{MIN|MAX|DEF}]
SOURce[1]:PULSe:HOLD {WIDTh|DCYClE}
SOURce[1]:PULSe:HOLD?

Description

The SOURce[1]:PULSe:PERiod command sets the period time of the internal modulator.

The SOURce[1]:PULSe:WIDTh command sets the pulse width of the internal modulator.

The SOURce[1]:PULSe:DCYClE command sets the duty cycle of the internal modulator.

The SOURce[1]:PULSe:HOLD command sets, for a pulsed waveform, the parameter to be held constant when the period changes.

Example

- SOUR:PULS:PER 0.0001
Sets the pulse period to 100µs.
- SOUR:PULS:DCYC MIN
Sets the duty cycle of the internal modulation to its minimum.
- SOUR:PULS:HOLD?
Queries the parameter to be held constant for a pulsed waveform.
- ← WIDT
The response is either WIDT or DCYC.

Note

This command is valid for LDC4000 and ITC4000 Series instruments only.

3.8 PD Sense Commands

The `SENSe[1]` subsystem controls the photodiode (PD) sense part of the instrument.

Note

These commands are valid for LDC4000 and ITC4000 Series instruments only. For LDC4000 and ITC4000 Series instruments the command suffix is 1 (can be omitted).

Table <%TABLENO++%>: PD Sense Commands

Command	Description
<code>SENSe[1]</code>	Path to PD sense subsystem
<code>[:CURRent][:DC]</code>	
<code>:RANGe[:UPPer]</code> ^[45] {MIN MAX <amps>}	Set photodiode range
<code>:RANGe[:UPPer]?</code> ^[45] [{MIN MAX}]	Return photodiode range
<code>:PROTection</code> ^[46]	
<code>[:LEVel]</code> {MIN MAX DEF <amps>}	Set photodiode current limit value
<code>[:LEVel]?</code> [{MIN MAX DEF}]	Return photodiode current limit value
<code>:TRIPed?</code>	Return protection tripped
<code>:CORRECTION</code>	
<code>:POWer[:PDIode][:RESPOnse]</code> ^[46] {MIN MAX DEF <amps>}	Set photodiode responsivity value in A/W
<code>:POWer[:PDIode][:RESPOnse]?</code> ^[46] [{MIN MAX DEF}]	Return photodiode responsivity value in A/W
<code>:POWer[:DC]</code>	
<code>:RANGe[:UPPer]</code> ^[47] {MIN MAX <watts>}	Set photodiode range in units of power
<code>:RANGe[:UPPer]?</code> ^[47] [{MIN MAX}]	Return photodiode range setting in units of power
<code>:PROTection</code> ^[47]	
<code>[:LEVel]</code> {MIN MAX DEF <watts>}	Set power limit value (photodiode)
<code>[:LEVel]?</code> [{MIN MAX DEF}]	Return power limit value (photodiode)
<code>:TRIPped</code>	Return protection tripped

3.8.1 Photodiode Current Range

Syntax

`SENSe[1][:CURRent][:DC]:RANGe[:UPPer]` {MIN|MAX|<amps>}

`SENSe[1][:CURRent][:DC]:RANGe[:UPPer]?` [{MIN|MAX}]

Description

The `SENSe[1][:CURRent][:DC]:RANGe[:UPPer]` command sets the photodiode current range

to the most appropriate range for the given value.

Example

→ SENS:RANG 1.7mA

Sets the photodiode current range to 2 mA (this is valid for 4000 Series instruments with a 2 mA photodiode input range).

→ SENS:RANG? MAX

Queries the biggest available photodiode current range.

← 2.000000E-02

Typical response.

Note

This command is valid for LDC4000 and ITC4000 Series instruments only.

3.8.2 Photodiode Current Protection

Syntax

SENSe[1][:CURRENT][:DC]:PROTection[:LEVel] {MIN|MAX|DEF|<amps>}

SENSe[1][:CURRENT][:DC]:PROTection[:LEVel]? [{MIN|MAX|DEF}]

SENSe[1][:CURRENT][:DC]:PROTection:TRIPped?

Description

The SENSe[1]:CURRENT[:DC]:PROTection[:LEVel] command sets the photodiode current limit value.

The SENSe[1][:CURRENT][:DC]:PROTection:TRIPped? query command returns an ASCII 1 if the photodiode current limit value is currently reached or an ASCII 0 if the photodiode current is currently below that limit.

Example

→ SENS:PROT 0.3mA

Sets the photodiode current limit to 0.3 mA.

Note

This command is valid for LDC4000 and ITC4000 Series instruments only.

3.8.3 Photodiode Responsivity

Syntax

SENSe[1][:CURRENT][:DC]:CORRection:POWeR[:PDIode][:RESPonse]
{MIN|MAX|DEF|<amps>}

SENSe[1][:CURRENT][:DC]:CORRection:POWeR[:PDIode][:RESPonse]?
{MIN|MAX|DEF}

Description

The SENSe[1][:CURRENT][:DC]:CORRection:POWeR[:PDIode][:RESPonse] command sets the photodiode responsivity in A/W .

Example

→ SENS:CORR:POW 511mA

Sets the photodiode responsivity to 511.0 mA/W .

Note

This command is valid for LDC4000 and ITC4000 Series instruments only.

3.8.4 Photodiode Power Range**Syntax**

```
SENSE[1]:POWER[:DC]:RANGE[:UPPER] {MIN|MAX|<watts>}
SENSE[1]:POWER[:DC]:RANGE[:UPPER]? [{MIN|MAX}]
```

Description

The `SENSE[1]:POWER[:DC]:RANGE[:UPPER]` command sets the power range to the most appropriate range for the given value.

Example

- `SENS:POW:RANG 0.012`
Sets power range to or above 12 mW depending on the set responsivity of the used photodiode.
- `SENS:POW:RANG? MAX`
Queries the biggest available power range.
- ← `1.800000E+00`
Typical response.

Note

This command is valid for LDC4000 and ITC4000 Series instruments only.

3.8.5 Photodiode Power Protection**Syntax**

```
SENSE[1]:POWER[:DC]:PROTECTION[:LEVEL] {MIN|MAX|DEF|<watts>}
SENSE[1]:POWER[:DC]:PROTECTION[:LEVEL]? [{MIN|MAX|DEF}]
SENSE[1]:POWER[:DC]:PROTECTION:TRIPPED?
```

Description

The `SENSE[1]:POWER[:DC]:PROTECTION[:LEVEL]` command sets the power limit value.

The `SENSE[1]:POWER[:DC]:PROTECTION:TRIPPED?` query command returns an ASCII 1 if the power limit value is currently reached or an ASCII 0 if the power is currently below that limit.

Example

- `SENS:POW:PROT 0.035`
Sets the power limit to 35.0 mW.

Note

This command is valid for LDC4000 and ITC4000 Series instruments only.

3.9 Thermopile/Power Meter Sense Commands

The `SENSE2` subsystem controls the thermopile/power meter sense part of the instrument.

Note

These commands are valid for LDC4000 and ITC4000 Series instruments only.

Table <TABLENO++>: Thermopile/Power Meter Sense Commands

Command	Description
SENSe2	Path to thermopile/power meter sense subsystem
[:VOLTage][:DC]	
:RANGe[:UPPer] ⁴⁸ {MIN MAX <amps>}	Set thermopile/power meter range
:RANGe[:UPPer]? ⁴⁸ [{MIN MAX}]	Return thermopile/power meter range
:PROTection ⁴⁹	
[:LEVel] {MIN MAX DEF <amps>}	Set thermopile/power meter voltage limit value
[:LEVel]? [{MIN MAX DEF}]	Return thermopile/power meter voltage limit value
:TRIPed?	Return protection tripped
:CORRECTION ⁴⁹	
:POWer[:PDIOde][:RESPonse] {MIN MAX DEF <amps>}	Set thermopile/power meter responsivity value in V/W
:POWer[:PDIOde][:RESPonse]? [{MIN MAX DEF}]	Return thermopile/power meter responsivity value in V/W
:POWer[:DC]	
:RANGe[:UPPer] ⁵⁰ {MIN MAX <watts>}	Set thermopile/power meter range in units of power
:RANGe[:UPPer]? ⁵⁰ [{MIN MAX}]	Return thermopile/power meter range setting in units of power
:PROTection ⁵⁰	
[:LEVel] {MIN MAX DEF <watts>}	Set power limit value (thermopile/power meter)
[:LEVel]? [{MIN MAX DEF}]	Return power limit value (thermopile/power meter)
:TRIPped	Return protection tripped

3.9.1 Thermopile/Power Meter Voltage Range

Syntax

```
SENSe2[:VOLTage][:DC]:RANGe[:UPPer] {MIN|MAX|<volts>}
SENSe2[:VOLTage][:DC]:RANGe[:UPPer]? [ {MIN|MAX} ]
```

Description

The `SENSe2[:VOLTage][:DC]:RANGe[:UPPer]` command sets the thermopile/power meter voltage range to the most appropriate range for the given value.

Example

- SENS2:RANG 0.5
Sets the thermopile/power meter voltage range to 1.0 V (this is valid for 4000 Series instruments with a 1 V thermopile/power meter input range).
- SENS2:RANG? MAX
Queries the biggest available thermopile/power meter voltage range.
- ← 1.000000E+01
Typical response.

Note

This command is valid for LDC4000 and ITC4000 Series instruments only.

3.9.2 Thermopile/Power Meter Voltage Protection**Syntax**

```
SENSe2[:VOLTage][:DC]:PROTection[:LEVel] {MIN|MAX|DEF|<volts>}
SENSe2[:VOLTage][:DC]:PROTection[:LEVel]? [{MIN|MAX|DEF}]
SENSe2[:VOLTage][:DC]:PROTection:TRIPped?
```

Description

The `SENSe2[:VOLTage][:DC]:PROTection[:LEVel]` command sets the thermopile/power meter voltage limit value.

The `SENSe2[:VOLTage][:DC]:PROTection:TRIPped?` query command returns an ASCII 1 if the thermopile/power meter voltage limit value is currently reached or an ASCII 0 if the thermopile/power meter voltage is currently below that limit.

Example

- SENS2:PROT 0.4V
Sets the thermopile/power meter voltage limit to 400 mV.

Note

This command is valid for LDC4000 and ITC4000 Series instruments only.

3.9.3 Thermopile/Power Meter Responsivity**Syntax**

```
SENSe2[:VOLTage][:DC]:CORRection:POWer[:THERmopile][:RESPonse]
{MIN|MAX|DEF|<volts>}
SENSe2[:VOLTage][:DC]:CORRection:POWer[:THERmopile][:RESPonse]?
{MIN|MAX|DEF}
```

Description

The `SENSe2[:VOLTage][:DC]:CORRection:POWer[:THERmopile][:RESPonse]` command sets the thermopile/power meter responsivity in V/W .

Example

- SENS2:CORR:POW 0.04V
Sets the thermopile/power meter responsivity to 40.0 mV/W .

Note

This command is valid for LDC4000 and ITC4000 Series instruments only.

3.9.4 Thermopile/Power Meter Power Range

Syntax

```
SENSe2:POWer[:DC]:RANGe[:UPPer] {MIN|MAX|<watts>}  
SENSe2:POWer[:DC]:RANGe[:UPPer]? [{MIN|MAX}]
```

Description

The `SENSe2:POWer[:DC]:RANGe[:UPPer] {MIN|MAX|<watts>}` command sets the thermopile/power meter power range to the most appropriate range for the given value.

Example

- `SENS2:POW:RANG 0.012`
Sets the power range to or above 12 mW depending on the set responsivity of the used thermopile/power meter.
- `SENS2:POW:RANG? MAX`
Queries the biggest available power range.
- ← `1.800000E+00`
Typical response.

Note

This command is valid for LDC4000 and ITC4000 Series instruments only.

3.9.5 Thermopile/Power Meter Power Protection

Syntax

```
SENSe2:POWer[:DC]:PROTection[:LEVel] {MIN|MAX|DEF|<watts>}  
SENSe2:POWer[:DC]:PROTection[:LEVel]? [{MIN|MAX|DEF}]  
SENSe2:POWer[:DC]:PROTection:TRIPped?
```

Description

The `SENSe2:POWer[:DC]:PROTection[:LEVel]` command sets the power limit value.

The `SENSe2:POWer[:DC]:PROTection:TRIPped?` query command returns an ASCII 1 if the power limit value is currently reached or an ASCII 0 if the power is currently below that limit.

Example

- `SENS:POW:PROT 0.511`
Sets the power limit to 511.0 mW.

Note

This command is valid for LDC4000 and ITC4000 Series instruments only.

3.10 QCW Pulse Trigger Commands

The `TRIGger[1]` subsystem controls the QCW mode triggering of the instrument.

Note

These commands are valid for LDC4000 and ITC4000 Series instruments only. For LDC4000 and ITC4000 Series instruments the command suffix is 1 (can be omitted).

Table <%TABLENO++%>: QCW Pulse Trigger Commands

Command	Description
<code>TRIGger</code>	Path to trigger functions
<code>[:SEquence] :SOURce^[51] { INTernal EXTernal }</code>	Set the trigger source
<code>TRIGger [:SEquence] :SOURce?^[51]</code>	Return the trigger source

3.10.1 QCW Trigger Source

Syntax

```
TRIGger [ :SEquence ] :SOURce { INTernal | EXTernal }
```

```
TRIGger [ :SEquence ] :SOURce?
```

Description

The `TRIGger [:SEquence] :SOURce` command sets the source for the QCW pulse trigger to either internal pulse generator or external pulse source connected through a rear panel BNC connector.

Example

- `TRIG:SOUR INT`
Sets the trigger source for QCW pulses to the internal pulse generator.
- `TRIG:SOUR EXT`
Sets the trigger source for QCW pulses to the rear panel BNC connector.
- `TRIG:SOUR?`
Queries the trigger source for QCW pulses.
- ← `INT`
The response is either `INT` or `EXT`.

Note

This command is valid for LDC4000 and ITC4000 Series instruments only.

3.11 TEC Source Commands

The `SOURCE{[1]|2}` subsystem controls the TEC source part of the instrument.

Note

These commands are valid for TED4000 and ITC4000 Series instruments only. For TED4000 Series instruments the command suffix is 1 (can be omitted), for ITC4000 Series instruments the suffix is 2.

Table <%TABLENO++%>: TEC Source Commands

Command	Description
<code>SOURCE{[1] 2}</code>	Path to TED source function
<code>:FUNCTION^[53]</code>	
<code>[:MODE]^[53] {TEMPERature CURREnt}</code>	Set function
<code>:CURREnt</code>	
<code>:LIMit[:AMPLitude]^[54] {MIN MAX <amps>}</code>	Set limit current value
<code>:LIMit[:AMPLitude]^[54] [{MIN MAX}]</code>	Return limit current value
<code>[:LEVel[:IMMediate][:AMPLitude]^[54] {MIN MAX <amps>}</code>	Set current setpoint
<code>[:LEVel[:IMMediate][:AMPLitude]?^[54] [{MIN MAX}]</code>	Return current setpoint
<code>:TEMPERature</code>	
<code>[:SPOint]^[55] {MIN MAX DEF <temperature>}</code>	Set temperature setpoint
<code>:LIMit^[55]</code>	
<code>:HIGH^[55] {MIN MAX <temperature>}</code>	Set settable temperature high limit
<code>:LOW^[55] {MIN MAX <temperature>}</code>	Set settable temperature low limit
<code>:LCONstants^[56]</code>	
<code>[:GAIN]^[56] {MIN MAX DEF <value>}</code>	Set PID loop P value
<code>:INTEgral^[56] {MIN MAX DEF <value>}</code>	Set PID loop I value
<code>:DERivative^[56] {MIN MAX DEF <value>}</code>	Set PID loop D value
<code>:PERiod^[56] {MIN MAX DEF <seconds>}</code>	Set thermal load oscillation period
<code>:ATUNE^[56]</code>	
<code>[:INITiate]^[56]</code>	Start AUTO-PID tune
<code>:CANCEl^[56]</code>	Cancel AUTO-PID tune
<code>:STATe?^[56]</code>	Query AUTO-PID state
<code>:LCONstants^[56]</code>	
<code>[:GAIN]?^[56]</code>	Query AUTO-PID P value

Command	Description
:INTegral? ^[56]	Query AUTO-PID I value
:DERivative? ^[56]	Query AUTO-PID D value
:PERiod? ^[56]	Query AUTO-PID oscillation period
:TRANsfer ^[56]	Transfer AUTO-PID to control loop
:PROTection ^[62]	
:DELay ^[62] {MIN MAX DEF <seconds>}	Set protection delay
:WINDow[:AMPLitude] ^[62] {MIN MAX DEF <temperature>}	Set temperature window amplitude
:TRIPped? ^[62]	Query protection tripped

3.11.1 TEC Source Function

Syntax

```
SOURce{[1]|2}:FUNction[:MODE] {TEMPerature|CURRENT}
SOURce{[1]|2}:FUNction[:MODE]?
```

Description

The `SOURce{[1]|2}:FUNction[:MODE]` command sets the instrument's TEC source function to current source mode or temperature control mode.

Example

```
→ SOUR:FUNC TEMP
```

Sets the TEC source function of a TED4000 Series instrument to temperature source mode.

Note

This command is valid for TED4000 and ITC4000 Series instruments only. For TED4000 Series instruments the command suffix is 1 (can be omitted), for ITC4000 Series instruments the suffix is 2.

3.11.2 TEC Current Limit

Syntax

```
SOURce{[1]|2}:CURRent:LIMit[:AMPLitude] {MIN|MAX|<amps>}  
SOURce{[1]|2}:CURRent:LIMit[:AMPLitude]? [{MIN|MAX}]
```

Description

The `SOURce{[1]|2}:CURRent:LIMit[:AMPLitude]` command sets the TEC element limit current in amperes.

Example

→ `SOUR:CURR:LIM 1.5`

Sets the TEC element limit current to 1.5A on a TED4000 Series instrument.

`SOUR2:CURR:LIM? MAX`

Queries the maximum settable TEC element limit current of a ITC4000 Series instrument.

← `1.500000E+01`

Typical response.

Note

This command is valid for TED4000 and ITC4000 Series instruments only. For TED4000 Series instruments the command suffix is 1 (can be omitted), for ITC4000 Series instruments the suffix is 2.

3.11.3 TEC Current Setpoint

Syntax

```
SOURce{[1]|2}:CURRent[:LEVel][:IMMediate][:AMPLitude]  
{MIN|MAX|<amps>}  
SOURce{[1]|2}:CURRent[:LEVel][:IMMediate][:AMPLitude]?  
[{MIN|MAX}]
```

Description

The `SOURce{[1]|2}:CURRent[:LEVel][:IMMediate][:AMPLitude]` command sets the TEC current setpoint in amperes in current source mode.

Example

→ `SOUR:CURR 1.0`

Sets the TEC element current to 1.0 A on a TED4000 Series instrument.

→ `SOUR2:CURR? MAX`

Queries the maximum settable TEC element current setpoint of a ITC4000 Series instrument.

← `1.500000E+01`

Typical response.

Note

This command is valid for TED4000 and ITC4000 Series instruments only. For TED4000 Series instruments the command suffix is 1 (can be omitted), for ITC4000 Series instruments the suffix is 2.

3.11.4 TEC Temperature Setpoint Limit

Syntax

```
SOURce{[1]|2}:TEMPerature:LIMit:HIGH {MIN|MAX|<temperature>}
SOURce{[1]|2}:TEMPerature:LIMit:HIGH? [{MIN|MAX}]
SOURce{[1]|2}:TEMPerature:LIMit:LOW {MIN|MAX|<temperature>}
SOURce{[1]|2}:TEMPerature:LIMit:LOW? [{MIN|MAX}]
```

Description

The `SOURce{[1]|2}:TEMPerature:LIMit` commands set a user defined temperature setpoint range in the selected temperature unit.

Example

```
→ SOUR:TEMP:LIM:LOW 0;HIGH 70
    Sets the temperature setpoint range to 0 ... 70 on a TED4000 Series instrument.
```

Note

This command is valid for TED4000 and ITC4000 Series instruments only. For TED4000 Series instruments the command suffix is 1 (can be omitted), for ITC4000 Series instruments the suffix is 2.

3.11.5 TEC Temperature Setpoint

Syntax

```
SOURce{[1]|2}:TEMPerature[:SPOint] {MIN|MAX|DEF|<temperature>}
SOURce{[1]|2}:TEMPerature[:SPOint]? [{MIN|MAX|DEF}]
```

Description

The `SOURce{[1]|2}:TEMPerature[:SPOint]` command sets the temperature setpoint in the selected temperature unit in temperature control mode. The default value is 25.0°C.

Example

```
→ SOUR:TEMP 25C
    Sets the temperature setpoint to 25°C on a TED4000 Series instrument.

→ SOUR2:TEMP?
    Queries the temperature setpoint of a ITC4000 Series instrument.

← 2.500000E+01
    Typical response.
```

Note

This command is valid for TED4000 and ITC4000 Series instruments only. For TED4000 Series instruments the command suffix is 1 (can be omitted), for ITC4000 Series instruments the suffix is 2.

3.11.6 PID Loop Constants

Syntax

```
SOURCE{[1]|2}:TEMPERature:LCONstants[:GAIN]
  {MIN|MAX|DEF|<value>}
SOURCE{[1]|2}:TEMPERature:LCONstants[:GAIN]?
  [{MIN|MAX|DEF}]
SOURCE{[1]|2}:TEMPERature:LCONstants:INTEgral
  {MIN|MAX|DEF|<value>}
SOURCE{[1]|2}:TEMPERature:LCONstants:INTEgral?
  [{MIN|MAX|DEF}]
SOURCE{[1]|2}:TEMPERature:LCONstants:DERivative
  {MIN|MAX|DEF|<value>}
SOURCE{[1]|2}:TEMPERature:LCONstants:DERivative?
  [{MIN|MAX|DEF}]
SOURCE{[1]|2}:TEMPERature:LCONstants:PERiod
  {MIN|MAX|DEF|<value>}
SOURCE{[1]|2}:TEMPERature:LCONstants:PERiod?
  [{MIN|MAX|DEF}]
```

Description

The `SOURCE{[1]|2}:TEMPERature:LCONstants` commands control the instrument PID temperature control loop. The gain value (proportional) is in $[\text{A}/\text{K}]$, its default value is 1.0. The integral value is in $[\text{A}/\text{K}\cdot\text{s}]$, its default value is 0.1. The derivative value is in $[\text{A}\cdot\text{s}/\text{K}]$, its default value is 0. The period value specifies the thermal load oscillation period in seconds [s], its default value is 1s.

Example

```
→ SOUR:TEMP:LCON:GAIN 1.0;INT 0.1;DER 0.0;PER 1.0
   Sets the PID control loop constants on a TED4000 Series instrument.
```

Note

This command is valid for TED4000 and ITC4000 Series instruments only. For TED4000 Series instruments the command suffix is 1 (can be omitted), for ITC4000 Series instruments the suffix is 2.

3.11.7 PID Auto Tune

Syntax

```
SOURCE{[1]|2}:TEMPERature:ATUNE[:INITiate]
SOURCE{[1]|2}:TEMPERature:ATUNE:CANCEL
SOURCE{[1]|2}:TEMPERature:ATUNE:STATE?
SOURCE{[1]|2}:TEMPERature:ATUNE:LCONstants[:GAIN]?
SOURCE{[1]|2}:TEMPERature:ATUNE:LCONstants:INTEgral?
SOURCE{[1]|2}:TEMPERature:ATUNE:LCONstants:DERivative?
SOURCE{[1]|2}:TEMPERature:ATUNE:LCONstants:PERiod?
SOURCE{[1]|2}:TEMPERature:ATUNE:LCONstants:TRANSFER
```

Description

The `SOURCE{[1]|2}:TEMPERature:ATUNE[:INITiate]` starts the Auto-PID procedure.

The `SOURCE{[1]|2}:TEMPERature:ATUNE:CANCEL` stops a running Auto-PID procedure.

The `SOURCE{[1]|2}:TEMPERature:ATUNE:STATE?` query retrieves information about a

previously run Auto-PID procedure.

The Auto-PID state query response is: state, phase, loop as numeric values, where state reflects the current state, phase shows the algorithm used and loop indicates the Auto-PID progress.

State	Description
0	Never run for this sensor/configuration
1	Auto-PID currently running
2	Canceled by user
3	Auto-PID failed
4	Auto-PID finished successfully

Phase	Description
0	Full tuning Auto-PID procedure
1	Fine tuning Auto-PID procedure

The query commands

```
SOURce{[1]|2}:TEMPerature:ATUNE:LCONstants[:GAIN]?
SOURce{[1]|2}:TEMPerature:ATUNE:LCONstants:INTEgral?
SOURce{[1]|2}:TEMPerature:ATUNE:LCONstants:DERivative?
SOURce{[1]|2}:TEMPerature:ATUNE:LCONstants:PERiod?
```

read back the evaluated Auto-PID loop constants.

You may pass the evaluated values to the PID temperature control loop using the command:

```
SOURce{[1]|2}:TEMPerature:ATUNE:LCONstants:TRANsfer
```

Note

This command is valid for TED4000 and ITC4000 Series instruments only. For TED4000 Series instruments the command suffix is 1 (can be omitted), for ITC4000 Series instruments the suffix is 2.

3.12 Temperature Sense Commands

The `SENSe{[1]|3}` subsystem controls the temperature sensing part of the instrument.

Note

These commands are valid for TED4000 and ITC4000 Series instruments only. For TED4000 Series instruments the command suffix is 1 (can be omitted), for ITC4000 Series instruments the suffix is 3.

Table <%TABLENO++%>: Temperature Sense Commands

Command	Description
<code>SENSe{[1] 3}</code>	Path to TEC sense function
<code>:TEMPerature</code>	
<code>:TRANsducer^[60][:TYPE] {AD590 THLow THHigh PT100 PT1000 LM35 LM335}</code>	Set temperature sensor type
<code>:TRANsducer^[60][:TYPE]?</code>	Return temperature sensor
<code>:THERmistor^[60]</code>	
<code>:METHod {EXPonential SHH}</code>	Set thermistor calculation method
<code>:METHod?</code>	Return thermistor calculation method
<code>[:SHH]</code>	
<code>:A {MIN MAX DEF <value>}</code>	Set Steinhart-Hart parameter A
<code>:A? [{MIN MAX DEF}]</code>	Return Steinhart-Hart parameter A
<code>:B {MIN MAX DEF <value>}</code>	Set Steinhart-Hart parameter B
<code>:B? [{MIN MAX DEF}]</code>	Return Steinhart-Hart parameter B
<code>:C {MIN MAX DEF <value>}</code>	Set Steinhart-Hart parameter C
<code>:C? [{MIN MAX DEF}]</code>	Return Steinhart-Hart parameter C
<code>:EXPonential</code>	
<code>:R0 {MIN MAX DEF <ohms>}</code>	Set exponential parameter R_0
<code>:R0? [{MIN MAX DEF}]</code>	Return exponential parameter R_0
<code>:T0 {MIN MAX DEF <temperature>}</code>	Set exponential parameter T_0
<code>:T0? [{MIN MAX DEF}]</code>	Return exponential parameter T_0
<code>:BETA {MIN MAX DEF <value>}</code>	Set exponential parameter beta
<code>:BETA? [{MIN MAX DEF}]</code>	Return exponential parameter beta
<code>:OFFSet^[61] {MIN MAX DEF <temperature>}</code>	Set transducer temperature offset

Command	Description
:OFFSet? ⁶¹ [{MIN MAX DEF}]	Return transducer temperature offset
:PROTection ⁶²	
:DELay ⁶² {MIN MAX DEF <seconds>}	Set protection reset delay
:DELay ⁶² ? {MIN MAX DEF}	Return protection reset delay
:WINDow[:AMPLitude] ⁶² {MIN MAX DEF <temperature>}	Set temperature window amplitude
:WINDow[:AMPLitude] ⁶² ? {MIN MAX DEF}	Return temperature window amplitude
:TRIPped? ⁶²	Return protection tripped

3.12.1 Sensor Selection

Syntax

```
SENSe{[1]|3}:TEMPerature:TRANsducer[:TYPE]  
  {AD590|THLow|THHigh|PT100|PT1000|LM35|LM335}  
SENSe{[1]|3}:TEMPerature:TRANsducer[:TYPE]?
```

Description

The `SENSe{[1]|3}:TEMPerature:TRANsducer[:TYPE]` command selects the temperature sensor to be used for future operation. See the operation manual for a more detailed description on temperature sensors for TED4000 and ITC4000 Series instruments.

Example

→ `SENS:TEMP:TRAN AD590`

Sets the temperature sensor to be used to an AD590 for a TED4000 Series instrument.

Note

This command is valid for TED4000 and ITC4000 Series instruments only. For TED4000 Series instruments the command suffix is 1 (can be omitted), for ITC4000 Series instruments the suffix is 3.

3.12.2 Thermistor Calculations

Syntax

```
SENSe{[1]|3}:TEMPerature:THERmistor:METHOD {EXPonential|SHH}  
SENSe{[1]|3}:TEMPerature:THERmistor:METHOD?
```

```
SENSe{[1]|3}:TEMPerature:THERmistor[:SHH]:A  
  {MIN|MAX|DEF|<value>}
```

```
SENSe{[1]|3}:TEMPerature:THERmistor[:SHH]:A?  
  [{MIN|MAX|DEF}]
```

```
SENSe{[1]|3}:TEMPerature:THERmistor[:SHH]:B  
  {MIN|MAX|DEF|<value>}
```

```
SENSe{[1]|3}:TEMPerature:THERmistor[:SHH]:B?  
  [{MIN|MAX|DEF}]
```

```
SENSe{[1]|3}:TEMPerature:THERmistor[:SHH]:C  
  {MIN|MAX|DEF|<value>}
```

```
SENSe{[1]|3}:TEMPerature:THERmistor[:SHH]:C?  
  [{MIN|MAX|DEF}]
```

```
SENSe{[1]|3}:TEMPerature:THERmistor:EXPonential:R0  
  {MIN|MAX|DEF|<value>}
```

```
SENSe{[1]|3}:TEMPerature:THERmistor:EXPonential:R0?  
  [{MIN|MAX|DEF}]
```

```
SENSe{[1]|3}:TEMPerature:THERmistor:EXPonential:T0  
  {MIN|MAX|DEF|<value>}
```

```
SENSe{[1]|3}:TEMPerature:THERmistor:EXPonential:T0?  
  [{MIN|MAX|DEF}]
```

```
SENSe{[1]|3}:TEMPerature:THERmistor:EXPonential:BETA  
  {MIN|MAX|DEF|<value>}
```

```
SENSe{[1]|3}:TEMPerature:THERmistor:EXPonential:BETA?  
  [{MIN|MAX|DEF}]
```

Description

The `SENSe{[1]|3}:TEMPerature:THERmistor:METHod` command selects one of the two temperature to/from resistance calculation methods (Steinhart-Hart or Exponential method).

The `SENSe{[1]|3}:TEMPerature:THERmistor[:SHH]:A` command sets the parameter A for the calculation method according to Steinhart-Hart.

The `SENSe{[1]|3}:TEMPerature:THERmistor[:SHH]:B` command sets the parameter B for the calculation method according to Steinhart-Hart.

The `SENSe{[1]|3}:TEMPerature:THERmistor[:SHH]:C` command sets the parameter C for the calculation method according to Steinhart-Hart.

The `SENSe{[1]|3}:TEMPerature:THERmistor:EXponential:R0` command sets the parameter R_0 (resistance at nominal temperature T_0) for the exponential calculation method.

The `SENSe{[1]|3}:TEMPerature:THERmistor:EXponential:T0` command sets the parameter T_0 (nominal temperature) for the exponential calculation method.

The `SENSe{[1]|3}:TEMPerature:THERmistor:EXponential:BETA` command sets the parameter Beta (also known as B-value) for the exponential calculation method.

Example

→ `SENS:TEMP:THER:METH EXP`

Sets the calculation method for a thermistor to exponential for a TED4000 Series instrument.

→ `SENS3:TEMP:THER:EXP:R0 10k;T0 25;BETA 3988`

Sets the three thermistor parameters needed (R_0 , T_0 , Beta) to calculate the temperature from a given resistance for a ITC4000 Series instrument.

Note

This command is valid for TED4000 and ITC4000 Series instruments only. For TED4000 Series instruments the command suffix is 1 (can be omitted), for ITC4000 Series instruments the suffix is 3.

3.12.3 Transducer Offset**Syntax**

`SENSe{[1]|3}:TEMPerature:OFFSet {MIN|MAX|DEF|<temperature>}`

`SENSe{[1]|3}:TEMPerature:OFFSet? [{MIN|MAX|DEF}]`

Description

The `SENSe{[1]|3}:TEMPerature:OFFSet` command sets a temperature offset for a temperature sensor (usually used to compensate small errors of temperature sensors).

Example

→ `SENS:TEMP:OFFS -0.2`

Sets a temperature offset of -0.2K for a TED4000 Series instrument.

Note

This command is valid for TED4000 and ITC4000 Series instruments only. For TED4000 Series instruments the command suffix is 1 (can be omitted), for ITC4000 Series instruments the suffix is 3.

3.12.4 Temperature Protection

Syntax

```
SENSE{[1]|3}:TEMPERature:PROTection:WINDow[:AMPLitude]
  {MIN|MAX|DEF|<temperature>}
SENSE{[1]|3}:TEMPERature:PROTection:WINDow[:AMPLitude]?
  [{MIN|MAX|DEF}]
SENSE{[1]|3}:TEMPERature:PROTection:DELAy {MIN|MAX|DEF|<seconds>}
SENSE{[1]|3}:TEMPERature:PROTection:DELAy? [{MIN|MAX|DEF}]
SENSE{[1]|3}:TEMPERature:PROTection:TRIPed?
```

Description

The `SENSE{[1]|3}:TEMPERature:PROTection:WINDow[:AMPLitude]` sets the temperature window width.

The `SENSE{[1]|3}:TEMPERature:PROTection:DELAy` sets the temperature window protection reset delay.

The `SENSE{[1]|3}:TEMPERature:PROTection:TRIPed?` query command returns an ASCII 1 if the window protection is currently active or an ASCII 0 if it's inactive.

Example

```
→ SENS:TEMP:PROT:WIND 1.5K;DEL 5
   Sets the temperature window amplitude to ±1.5 Kelvin and the protection reset delay to 5 seconds for a TED4000 Series instrument.
```

Note

This command is valid for TED4000 and ITC4000 Series instruments only. For TED4000 Series instruments the command suffix is 1 (can be omitted), for ITC4000 Series instruments the suffix is 3.

See also [Temperature Protection Mode](#)^[31] to protect the LD output of an ITC4000 Series instrument.

3.13 TEC Output Commands

The `OUTPut{[1]|2}` subsystem controls the TEC output of the instrument.

Note

These commands are valid for TED4000 and ITC4000 Series instruments only. For TED4000 Series instruments the command suffix is 1 (can be omitted), for ITC4000 Series instruments the suffix is 2.

Table <%TABLENO++%>: TEC Output Commands

Command	Description
<code>OUTPut{[1] 2}</code>	Path to TEC output function
<code>[:STATe]^[63] {ON 1 OFF 0}</code>	Enable (ON) or disable (OFF) TEC output
<code>[:STATe]?^[63]</code>	Return output state
<code>:PROTection</code>	Path to TEC output protection
<code>:CABLe^[64]</code>	Path to cable failure detection
<code>:TRIPped?</code>	Return protection tripped
<code>:TRANsducer^[64]</code>	Path to temperature sensor protection
<code>:TRIPped?</code>	Return protection tripped
<code>:OTEMp^[64]</code>	Path to Over Temperature protection circuit
<code>:TRIPped?</code>	Return protection tripped

3.13.1 Output State

Syntax

```
OUTPut{[1]|2}[ :STATe ] {OFF|0|ON|1}
```

```
OUTPut{[1]|2}[ :STATe ]?
```

Description

The `OUTPut{[1]|2}[:STATe]` command switches the instrument's TEC output on/off.

Example

- `OUTP ON`
Switches the TEC output of a TED4000 Series instrument on.
- `OUTP?`
Queries the state of the TEC output of a TED4000 Series instrument.
- `OUTP2 ON`
Switches the TEC output of an ITC4000 Series instrument on.
- `OUTP2?`
Queries the state of the TEC output of an ITC4000 Series instrument.
- ← `1`
The response is ASCII `1` if the output is switched on or an ASCII `0` if the output is off.

Note

The `*RST[11]` command switches the output off.

Note

These commands are valid for TED4000 and ITC4000 Series instruments only. For TED4000 Series instruments the command suffix is 1 (can be omitted), for ITC4000 Series instruments the suffix is 2.

3.13.2 Protection Queries

Syntax

OUTPut[1]:PROTection:CABLe:TRIPped?

OUTPut[1]:PROTection:TRANsducer:TRIPped?

OUTPut[1]:PROTection:OTEMp:TRIPped?

Description

The `OUTPut[1]:PROTection:CABLe:TRIPped?` query command returns a 1 if the cable connection from pin 1 to pin 15 (interlock) of the TEC-DRIVER 17W2 D-Sub jack is disrupted and returns a 0 if the connection is closed.

The `OUTPut[1]:PROTection:TRANsducer:TRIPped?` query command returns 1 if a sensor failure condition is detected and a 0 if a sensor is detected.

The `OUTPut[1]:PROTection:OTEMp:TRIPped?` query command returns 1 if the internal temperature of the instrument is too hot and a 0 when not.

Example

- `OUTP:PROT:CABL:TRIP?`
Queries the cable failure protection of the TEC part of the instrument.
- ← 0
Typical response.

Note


These commands are valid for TED4000 and ITC4000 only.

3.14 Status Subsystem Commands

The Status Subsystem of the 4000 Series instruments is implemented based on the SCPI 1999.0 standard chapter 9 'Status Reporting' and the status mechanism described in chapter 11 of IEEE 488.2 (see also [IEEE-488.2 commands](#)^[10]).

Table <%TABLENO++%>: Status Commands

Command	Description
STATus	Path to status subsystem
:AUXiliary ^[69]	Path to control auxiliary event registers
[:EVENT]?	Read the event register
:CONDition?	Read the condition register
:PTRansition <value>	Program the positive transition filter
:PTRansition?	Read the positive transition filter
:NTRansition <value>	Program the negative transition filter
:NTRansition?	Read the negative transition filter
:ENABle <value>	Program the enable register
:ENABle?	Read the enable register
:MEASurement ^[70]	Path to measurement event registers
[:EVENT]?	Read the event register
:CONDition?	Read the condition register
:PTRansition <value>	Program the positive transition filter
:PTRansition?	Read the positive transition filter
:NTRansition <value>	Program the negative transition filter
:NTRansition?	Read the negative transition filter
:ENABle <value>	Program the enable register
:ENABle?	Read the enable register
:QUESTionable ^[71]	Path to control questionable event registers
[:EVENT]?	Read the event register
:CONDition?	Read the condition register
:PTRansition <value>	Program the positive transition filter
:PTRansition?	Read the positive transition filter
:NTRansition <value>	Program the negative transition filter
:NTRansition?	Read the negative transition filter
:ENABle <value>	Program the enable register
:ENABle?	Read the enable register
:OPERation ^[72]	Path to control operation event registers
[:EVENT]?	Read the event register
:CONDition?	Read the condition register
:PTRansition <value>	Program the positive transition filter

Command	Description
:PTRansition?	Read the positive transition filter
:NTRansition <value>	Program the negative transition filter
:NTRansition?	Read the negative transition filter
:ENABle <value>	Program the enable register
:ENABle?	Read the enable register
:PRESet  ⁷³	Return status registers to default states

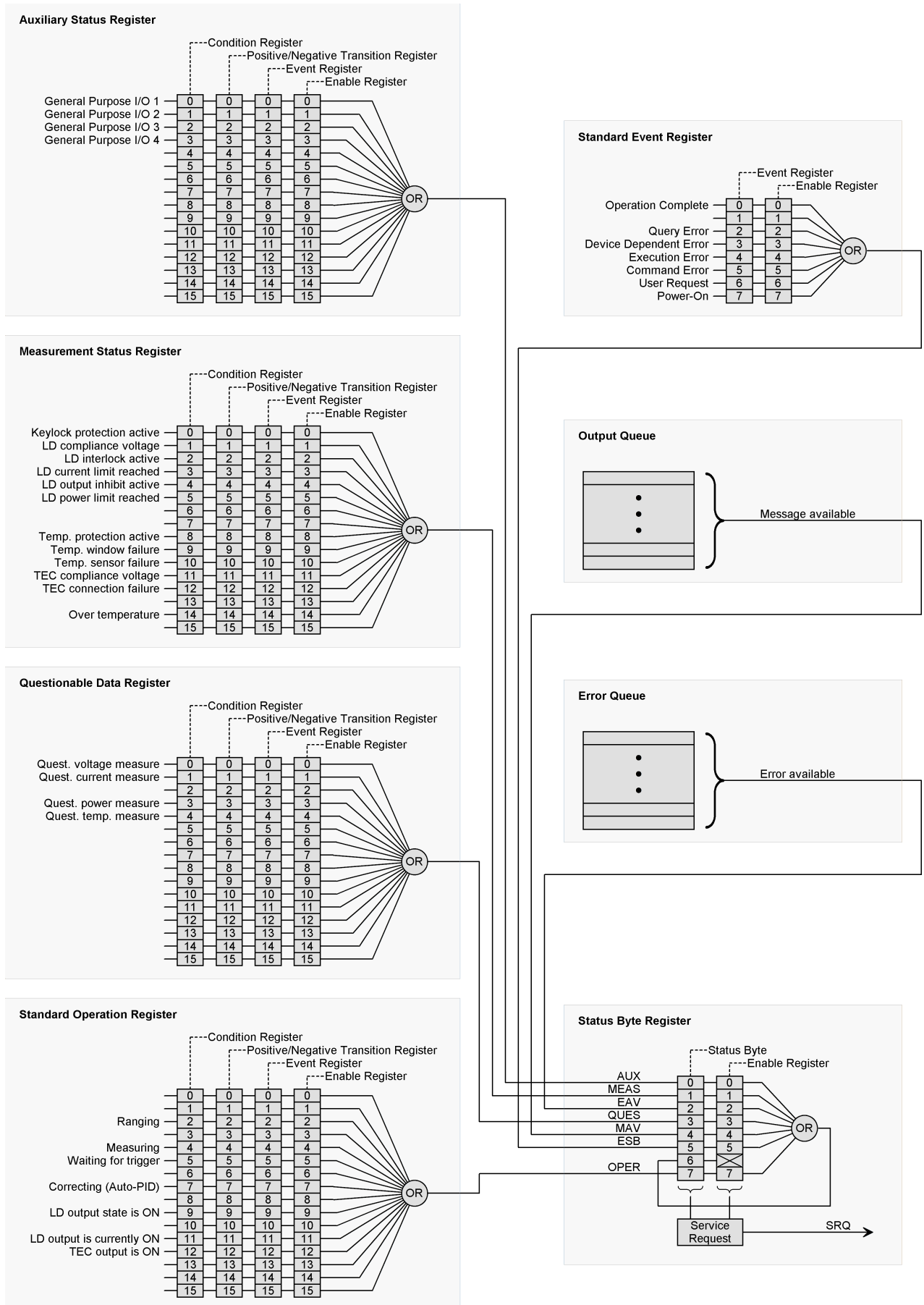


Figure <%FIGURENO++%>: Status Structure

3.14.1 Programming and Reading Registers

The status system records various instrument conditions and states in several register groups. Each register group is made up of several low-level registers called the condition register, event register, filter register and enable register which control the action of specific bits within the register group.

A **condition register** continuously monitors the state of the instrument. The bits in the condition register are updated in real-time and the bits are not latched or buffered. This is a read-only register and the bits are not cleared when you read the register.

An **event register** latches the various events from the changes in the condition register. There is no buffering in this register; while an event bit is set, subsequent events corresponding to that bit are ignored. This is a read-only register. Once a bit is set, it remains set until cleared by a query or clear status (`*CLS`^[14]) command.

An **enable register** defines which bits in the event register will be reported to the *Status Byte Register Group*. You can write to or read from an enable register.

A **filter register** (= positive/negative transition register) defines which transition of the condition register will cause an event for the event register. A set bit in the positive (negative) transition register will lead to a set bit in the event register when a low to high (high to low) transition of the corresponding bit in the condition register occurs.

The only registers that can be programmed by the user are the enable and the filter registers. All other registers in the status structure are read-only registers. The following explains how to get the parameter values for the various commands used to program enable registers.

A command to program an event enable register or a filter register is sent with a parameter value that determines the desired state (0 or 1) of each bit in the appropriate register. An enable register or a filter register can be programmed using binary, decimal, hexadecimal or octal data formats for the parameter value (for a detailed description of numeric formats see chapter 7.7.4 <NONDECIMAL NUMERIC PROGRAM DATA> of IEEE488.2-1992).

Example

The four commands program the *Auxiliary Enable Register* to the same value:

- `STAT:AUX:ENAB 2081`
Program the *Auxiliary Enable Register* with 2018 decimal.
- `STAT:AUX:ENAB #H821`
Program the *Auxiliary Enable Register* with 821 hexadecimal.
- `STAT:AUX:ENAB #Q4041`
Program the *Auxiliary Enable Register* with 4041 octal.
- `STAT:AUX:ENAB #B100000100001`
Program the *Auxiliary Enable Register* with 100000100001 binary.

The bit positions of the register (see table below) indicate the binary parameter value. For example, if you wish to set bits B11, B5, and B0, the binary value would be 100000100001 (where B11=1, B5=1, B0=1, and all other bits are 0). When you use one of the other formats, convert the binary number to its decimal, hexadecimal, or octal equivalent:

Binary 100000100001 = Decimal 2081 = Hexadecimal 821 = Octal 4041

Note that the table below includes the decimal weight for each register bit. To set bits B11, B5, and B0, the decimal parameter value would be the sum of the decimal weights for those bits (2048 + 32 + 1 = 2081).

Table <%TABLENO++%>: 16-bit Register Values

bit position	B7	B6	B5	B4	B3	B2	B1	B0
binary value	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1
decimal weights (as power of 2)	128 (2 ⁷)	64 (2 ⁶)	32 (2 ⁵)	16 (2 ⁴)	8 (2 ³)	4 (2 ²)	2 (2 ¹)	1 (2 ⁰)
bit position	B15	B14	B13	B12	B11	B10	B9	B8
binary value	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1
decimal weights (as power of 2)	32768 (2 ¹⁵)	16384 (2 ¹⁴)	8192 (2 ¹³)	4096 (2 ¹²)	2048 (2 ¹¹)	1024 (2 ¹⁰)	512 (2 ⁹)	256 (2 ⁸)

3.14.2 Auxiliary Register Group

Syntax

```

STATUS:AUXiliary[:EVENT]?
STATUS:AUXiliary:CONDition?
STATUS:AUXiliary:PTRansition <value>
STATUS:AUXiliary:PTRansition?
STATUS:AUXiliary:NTRansition <value>
STATUS:AUXiliary:NTRansition?
STATUS:AUXiliary:ENABle <value>
STATUS:AUXiliary:ENABle?

```

Description

The `STATUS:AUXiliary` commands and queries handle the registers of the *Auxiliary Register Group*.

Table <%TABLENO++%>: Auxiliary Register Group

Bit Number	Decimal Value	Definition
0	1	General Purpose I/O ₁
1	2	General Purpose I/O ₂
2	4	General Purpose I/O ₃
3	8	General Purpose I/O ₄
4-15		reserved

Note

For more information on the SCPI Status System for Thorlabs 4000 Series instruments, see chapter [Programming and Reading Registers](#)^[68].

3.14.3 Measurement Register Group

Syntax

```

STATus:MEASurement[:EVENT]?
STATus:MEASurement:CONDition?
STATus:MEASurement:PTRansition <value>
STATus:MEASurement:PTRansition?
STATus:MEASurement:NTRansition <value>
STATus:MEASurement:NTRansition?
STATus:MEASurement:ENABle <value>
STATus:MEASurement:ENABle?

```

Description

The `STATus:MEASurement` commands and queries handle the registers of the *Measurement Register Group*.

Table <%TABLENO++%>: Measurement Register Group

Bit Number	Decimal Value	Definition	
0	1	keylock protection active	
1	2	LD compliance voltage	LDC, ITC
2	4	LD interlock active	LDC, ITC
3	8	LD current limit reached	LDC, ITC
4	16	LD output inhibit active	LDC, ITC
5	32	LD power limit reached	LDC, ITC
6	64	not used	
7	128	not used	
8	256	Temperature protection active	TED, ITC
9	512	Temperature window failure	TED, ITC
10	1024	Temperature sensor failure	TED, ITC
11	2048	TEC compliance voltage reached	TED, ITC
12	4096	TEC connection failure	TED, ITC
13	8192	not used	
14	16384	Over temperature	
15	32768	reserved	

Note

For more information on the SCPI Status System for Thorlabs 4000 Series instruments, see chapter [Programming and Reading Registers](#) ^[68].

3.14.4 Questionable Register Group

Syntax

```

STATUS:QUESTIONable[:EVENT]?
STATUS:QUESTIONable:CONDition?
STATUS:QUESTIONable:PTRansition <value>
STATUS:QUESTIONable:PTRansition?
STATUS:QUESTIONable:NTRansition <value>
STATUS:QUESTIONable:NTRansition?
STATUS:QUESTIONable:ENABle <value>
STATUS:QUESTIONable:ENABle?

```

Description

The `STATUS:QUESTIONable` commands and queries handle the registers of the *Questionable Register Group*.

Table <%TABLENO++%>: Questionable Register Group

Bit Number	Decimal Value	Definition
0	1	Questionable voltage measure
1	2	Questionable current measure
2	4	not used
3	8	Questionable power measure
4	16	Questionable temperature measure
5-15		not used

Note

For more information on the SCPI Status System for Thorlabs 4000 Series instruments, see chapter [Programming and Reading Registers](#)⁶⁸.

3.14.5 Operation Registers Group

Syntax

```

STATus:OPERation:[ :EVENT]?
STATus:OPERation:CONDition?
STATus:OPERation:PTRansition <value>
STATus:OPERation:PTRansition?
STATus:OPERation:NTRansition <value>
STATus:OPERation:NTRansition?
STATus:OPERation:ENABle <value>
STATus:OPERation:ENABle?

```

Description

The `STATus:OPERation` commands and queries handle the registers of the *Operation Register Group*.

Table <%TABLENO++%>: Operation Register Group

Bit Number	Decimal Value	Definition	
0	1	not used	
1	2	not used	
2	4	Ranging	LDC, TED, ITC
3	8	not used	
4	16	Measuring	LDC, TED, ITC
5	32	Waiting for trigger	LDC, ITC
6	64	not used	
7	128	Correcting (Auto-PID)	TED, ITC
8	256	not used	
9	512	LD output state is ON	LDC, ITC
10	1024	not used	
11	2048	LD output is currently ON	LDC, ITC
12	4096	TEC output is ON	TED, ITC
13	8192	not used	
14	16384	not used	
15	32768	reserved	

Note

For more information on the SCPI Status System for Thorlabs 4000 Series instruments, see chapter [Programming and Reading Registers](#)⁶⁸.

3.14.6 Status Preset Command

Syntax

STATus:PRESet

Description

The STATus:PRESet command presets the registers of the [Auxiliary Register Group](#)^[69], the [Measurement Register Group](#)^[70], the [Questionable Register Group](#)^[71] and the [Operation Register Group](#)^[72] to the default values shown in the table below.

Table <%TABLENO++%>: Default Register Values

Register	Default Value
<i>Auxiliary Status Enable Register</i>	all set
<i>Auxiliary Status Positive Transition Register</i>	all set
<i>Auxiliary Status Negative Transition Register</i>	all cleared
<i>Measurement Status Enable Register</i>	all set
<i>Measurement Status Positive Transition Register</i>	all set
<i>Measurement Status Negative Transition Register</i>	all cleared
<i>Questionable Status Enable Register</i>	all cleared
<i>Questionable Status Positive Transition Register</i>	all set
<i>Questionable Status Negative Transition Register</i>	all cleared
<i>Standard Operation Status Enable Register</i>	all cleared
<i>Standard Operation Status Positive Transition Register</i>	all set
<i>Standard Operation Status Negative Transition Register</i>	all cleared

Note

For more information on the SCPI Status System for Thorlabs 4000 Series instruments, see chapter [Programming and Reading Registers](#)^[68].

3.15 Memory Subsystem Commands

The `MEMory` subsystem manages instrument memory.

Table <TABLENO++>: Memory Commands

Command	Description
<code>MEMory</code>	Path to memory subsystem
<code>:NSTates?</code> ^[74]	Return number of state memories
<code>:STATE</code>	
<code>:NAME</code> ^[74] <value>, <name>	Assign a name to a state memory
<code>:NAME?</code> ^[74] <value>	Return the name of a state memory

3.15.1 Number of State Memories

Syntax

`MEMory:NSTates?`

Description

The `MEMory:NSTates?` query command returns the number of state memories. For 4000 Series instruments the answer will be 8. See also chapter [*SAV, *RCL, *SDS - Save/Recall Setup](#) ^[12]

Example

```
→ MEM:NST?
   Queries the number of state memories available.

← 8
   Typical response.
```

3.15.2 State Memories Names

Syntax

`MEMory:STATE:NAME <value>, <name>`

`MEMory:STATE:NAME? <value>`

Description

The `MEMory:STATE:NAME` command assigns a name to a state memory. For the parameter <value> remember that the state memories 1 ... 8 are numbered 0 ... 7.

Example

```
→ MEM:STAT:NAME 0, "Experiment 5"
   Sets the name of state memory 1 to 'Experiment 5'.
```

3.16 Display Subsystem Commands

The `DISPlay` subsystem controls the front panel display (LCD) of a Thorlabs Series 4000 instrument.

Table <%%TABLENO++%>: Display Commands

Command	Description
<code>DISPlay</code>	Path to display subsystem
<code>:BRIGHTness</code> ⁷⁵ <value>	Set the display brightness
<code>:BRIGHTness?</code> ⁷⁵	Return the display brightness
<code>:CONTRast</code> ⁷⁵ <value>	Set the display contrast
<code>:CONTRast?</code> ⁷⁵	Return the display contrast

3.16.1 Display Brightness

Syntax

```
DISPlay:BRIGHTness <value>
```

```
DISPlay:BRIGHTness?
```

Description

The `DISPlay:BRIGHTness` command controls the intensity of the display. The range of the parameter is 0 to 1, where 1 is full intensity and 0 is fully blanked.

Example

```
→ DISP:BRIGHT 0.6
```

Sets the brightness of the front panel display to 60% of full intensity

3.16.2 Display Contrast

Syntax

```
DISPlay:CONTRast <value>
```

```
DISPlay:CONTRast?
```

Description

The `DISPlay:CONTRast` command determines the relative difference in brightness between “full” intensity and “no” intensity as displayed. The parameter ranges in value from 0 to 1; 0 indicates no difference between data and background, and 1 indicates maximum contrast.

Example

```
→ DISP:CONT 0.5
```

Sets the contrast of the front panel display to 50%.

3.17 Digital I/O Output Commands

The `SOURCE{2|3}` subsystem controls the instrument's digital I/O output.

Note

For LDC4000 and TED4000 Series instruments the command suffix is 2, for ITC4000 Series instruments the suffix is 3.

Table <TABLENO++%>: Digital I/O Output Commands

Command	Description
<code>SOURCE{2 3}</code>	Path to output functions
<code>:DIGital</code>	Path to digital I/O functions
<code>:DATA^[76] <value></code>	Set digital I/O output data
<code>:DATA?^[76]</code>	Return digital I/O output data
<code>:ENABle^[77] <value></code>	Set digital I/O output enable value
<code>:ENABle?^[77]</code>	Return digital I/O output enable value

3.17.1 Digital I/O Output

Syntax

`SOURCE{2|3}:DIGital:DATA <value>`

`SOURCE{2|3}:DIGital:DATA?`

Description

The `SOURCE{2|3}:DIGital:DATA` command sets the I/O pins according to value. 'Value' can be calculated as:

$$\text{value} = 1 \times I/O_1 + 2 \times I/O_2 + 4 \times I/O_3 + 8 \times I/O_4.$$

where:

$$\begin{aligned} I/O_n &= 0 \text{ for } I/O \text{ is low level,} \\ I/O_n &= 1 \text{ for } I/O \text{ is high level.} \end{aligned}$$

In table '[Digital I/O values](#)'^[78] all possible I/O values are listed in decoded form.

Example

→ `SOUR3:DIG:DATA 8`

Sets I/O_4 of the digital I/O port of an ITC4000 to high, I/O_1 , I/O_2 , I/O_3 to low.

Note

$I/O_{1...4}$ of the digital I/O port can be configured as input or output (see [Digital I/O configuration](#)^[77]). To drive an external load the respective I/O has to be configured as output.

Note

For TED4000 Series instruments and for LDC4000 Series instruments the command suffix is 2, for ITC4000 Series instruments the suffix is 3.

3.17.2 Digital I/O Configuration

Syntax

```
SOURce{2|3}:DIGital:ENABle <value>
SOURce{2|3}:DIGital:ENABle?
```

Description

The `SOURce{2|3}:DIGital:ENABle` command enables the I/O pins as output according to value. 'Value' can be calculated as:

$$\text{value} = 1 \times I/O_1 + 2 \times I/O_2 + 4 \times I/O_3 + 8 \times I/O_4.$$

where:

$$\begin{aligned} I/O_n &= 0 \text{ for I/O is input,} \\ I/O_n &= 1 \text{ for I/O is output.} \end{aligned}$$

In table ['Digital I/O values'](#)^[78] all possible I/O values are listed in decoded form.

Example

```
→ SOUR3:DIG:ENAB 15
   Sets I/O1...4 of the digital I/O port of an ITC4000 to output.
```

Note

For TED4000 Series instruments and for LDC4000 Series instruments the command suffix is 2, for ITC4000 Series instruments the suffix is 3.

3.18 Digital I/O Input Commands

The `SENSE{2|3|4}` subsystem controls the instrument's digital I/O input.

Note

For LDC4000 Series instruments the command suffix is 3, for TED4000 Series instruments the command suffix is 2 and for ITC4000 Series instruments the command suffix is 4.

Table <%TABLENO++%>: Digital I/O Input Commands

Command	Description
<code>SENSE{2 3 4}</code>	Path to input functions
<code>:DIGital</code>	Path to digital I/O functions
<code>:DATA?</code> ^[78]	Return digital I/O input data

3.18.1 Digital I/O Input

Syntax

SENSe{2|3|4}:DIGital:DATA?

Description

The SENSe{2|3|4}:DIGital:DATA? command queries the I/O pins of the digital I/O port. The returned value represents the levels of I/O_{1...4} according to:

Table <%TABLENO++%>: Digital I/O Values

I/O ₄	I/O ₃	I/O ₂	I/O ₁	value
L	L	L	L	0
L	L	L	H	1
L	L	H	L	2
L	L	H	H	3
L	H	L	L	4
L	H	L	H	5
L	H	H	L	6
L	H	H	H	7
H	L	L	L	8
H	L	L	H	9
H	L	H	L	10
H	L	H	H	11
H	H	L	L	12
H	H	L	H	13
H	H	H	L	14
H	H	H	H	15

Example

→ SENS2:DIG:DATA?

Queries the digital I/O port of a TED4000 Series instrument.

← 9

Typical response.

Note

I/O_{1...4} of the digital I/O port can be configured as input or output (see [Digital I/O configuration](#) [77]). To be able to read the level of a connected driver the respective I/O has to be configured as input.

Note

For LDC4000 Series instruments the command suffix is 3, for TED4000 Series instruments the command suffix is 2 and for ITC4000 Series instruments the command suffix is 4.

3.19 Unit Subsystem Commands

For each SCPI command default units are defined, where applicable. The `UNIT` subsystem provides a mechanism to change these default units. The units selected apply to the designated command parameters for both command and response

Table <%TABLENO++%>: Unit Commands

Command	Description
<code>UNIT</code>	Path to unit functions
<code>:TEMPerature</code> ⁷⁹ { <code>C</code> <code>CEL</code> <code>CELSius</code> <code>F</code> <code>FAR</code> <code>FAHRenheit</code> <code>K</code> <code>KELVin</code> }	Set the temperature unit
<code>:TEMPerature?</code> ⁷⁹	Return the temperature unit

3.19.1 Temperature Unit Commands

Syntax

```
UNIT:TEMPerature {C|CEL|CELSius|F|FAR|FAHRenheit|K|KELVin}
UNIT:TEMPerature?
```

Description

The `UNIT:TEMPerature` command sets the temperature unit used for further communication with the instrument.

Example

```
→ UNIT:TEMP K
```

Sets the temperature unit for further communication with the instrument to Kelvin.


Note

This command is valid only for TED4000 and ITC4000 Series instruments.

3.20 Calibration Subsystem Commands

4000 Series instruments do not provide an internal calibration mechanism, they are calibrated at the factory. The `CALibration` subsystem for 4000 Series instruments only provides a means of retrieving the instrument's calibration string. This string is factory set at calibration time.

Table <%TABLENO++%>: Calibration Commands

Command	Description
<code>CALibration</code>	Path to calibration functions
<code>:STRing?</code> 	Return the calibration string

3.20.1 Calibration String

Syntax

`CALibration:STRing?`

Description

The `CALibration:STRing?` command returns the calibration string.

Example

```
→ CAL:STR?
   Queries the calibration string.

← CALIBRATED JUNE-30-2009
   Typical response.
```

4 SCPI Error Messages

[Command Errors \(-100 ... -199\)](#)^[82]

[Execution Errors \(-200 ... -299\)](#)^[83]

[Device Specific Errors \(-300 ... -399\)](#)^[84]

[Query Errors \(-400 ... -499\)](#)^[84]

[Instrument Errors \(0...\)](#)^[85]

LDC4000, TED4000 and ITC4000 Series instruments return error messages in accord with the SCPI standard.

- A record of up to 10 errors can be stored in the instrument's error queue.
- The front-panel ERROR annunciator turns on when one or more errors are currently stored in the error queue.
- Errors are retrieved in first-in-first-out (FIFO) order. The first error returned is the first error that was stored.
- Errors are cleared as you read them. When you have read all errors from error queue, the ERROR annunciator turns off.
- If more than 10 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 read from the queue.
- If no errors have occurred when you read the error queue, the instrument responds with +0, "No error".
- Error conditions are also summarized in the Status Byte Register. For more information on the SCPI Status System for the LDC4000, TED4000 and ITC4000 Series instruments, see chapter [Status Subsystem Commands](#)^[65].
- The error queue is cleared by the [*CLS](#)^[14] (Clear Status) command and when power is cycled. All errors are cleared when you read the error queue. The error queue is not cleared by the Reset ([*RST](#)^[11] command) or an status preset command ([STATus:PRESet](#)^[73] command).
- For reading the error queue refer to chapter [Error Query](#)^[16].

4.1 Command Errors

Error	Description
-100	<i>Command error</i> This is the generic syntax error for devices that cannot detect more specific errors. This code indicates only that a Command Error as defined in IEEE 488.2, 11.5.1.1.4 has occurred.
-101	<i>Invalid character</i> A syntactic element contains a character which is invalid for that type; for example, a header containing an ampersand, SETUP&. This error might be used in place of errors -114, -121, -141, and perhaps some others.
-102	<i>Syntax error</i> An unrecognized command or data type was encountered; for example, a string was received when the device does not accept strings.
-103	<i>Invalid separator</i> The parser was expecting a separator and encountered an illegal character; for example, the semicolon was omitted after a program message unit, INP:POL CG*STB? instead of INP:POL CG;*STB?
-104	<i>Data type error</i> The parser recognized a data element different than one allowed; for example, numeric or string data was expected but block data was encountered.
-105	<i>GET not allowed</i> A Group Execute Trigger was received within a program message (see IEEE 488.2, 7.7).
-108	<i>Parameter not allowed</i> More parameters were received than expected for the header; for example, the *SRE common command only accepts one parameter, so receiving *SRE 0,1 is not allowed.
-109	<i>Missing parameter</i> Fewer parameters were received than required for the header; for example, the *SRE common command requires one parameter, so receiving *SRE is not allowed.
-110	<i>Command header error</i> An error was detected in the header. This error message should be used when the device cannot detect the more specific errors described for errors -111 through -119.
-113	<i>Undefined header (Unknown command)</i> The header is syntactically correct, but it is undefined for this specific device; for example, *XYZ is not defined for any device.
-115	<i>Unexpected number of parameters</i> The number of parameters received does not correspond to the number of parameters expected.
-120	<i>Numeric data error</i> This error is generated when parsing a data element which appears to be numeric, including the nondecimal numeric types.
-130	<i>Suffix error</i> This error, as well as error -131, is generated when parsing a suffix.
-131	<i>Invalid suffix</i> The suffix does not follow the syntax described in IEEE 488.2, 7.7.3.2, or the suffix is inappropriate for this device.
-151	<i>Invalid string data</i> A string data element was expected, but was invalid for some reason (see IEEE 488.2, 7.7.5.2); for example, an END message was received before the terminal quote character.

4.2 Execution Errors

Error	Description
-220	<i>Parameter error</i> Indicates that a program data element related error occurred.
-221	<i>Settings conflict</i> Indicates that a legal program data element was parsed but could not be executed due to the current device state (see IEEE 488.2, 6.4.5.3 and 11.5.1.1.5.)
-222	<i>Data out of range</i> Indicates that a legal program data element was parsed but could not be executed because the interpreted value was outside the legal range as defined by the device (see IEEE 488.2, 11.5.1.1.5.)
-230	<i>Data corrupt or stale</i> Possibly invalid data; new reading started but not completed since last access.
-231	<i>Data questionable</i> Indicates that measurement accuracy is suspect.
-240	<i>Hardware error</i> Indicates that a legal program command or query could not be executed because of a hardware problem in the device.
-241	<i>Hardware missing</i> Indicates that a legal program command or query could not be executed because of missing device hardware; for example, an option was not installed.
-250	<i>Mass storage error</i> Indicates that a mass storage error occurred.
-251	<i>Missing mass storage</i> Indicates that a legal program command or query could not be executed because of missing mass storage; for example, an option that was not installed.
-252	<i>Missing media</i> Indicates that a legal program command or query could not be executed because of a missing media; for example, no SD-card or no USB-stick.
-253	<i>Corrupt media</i> Indicates that a legal program command or query could not be executed because of corrupt media; for example, bad SD-card or wrong format.
-254	<i>Media full</i> Indicates that a legal program command or query could not be executed because the media was full; for example, there is no room on the SD-card.
-255	<i>Directory full</i> Indicates that a legal program command or query could not be executed because the media directory was full.
-256	<i>File name not found</i> Indicates that a legal program command or query could not be executed because the file name on the device media was not found; for example, an attempt was made to read or copy a nonexistent file.
-257	<i>File name error</i> Indicates that a legal program command or query could not be executed because the file name on the device media was in error; for example, an attempt was made to copy to a duplicate file name.
-258	<i>Media protected</i> Indicates that a legal program command or query could not be executed because the media was protected; for example, the write-protect tab on a disk was present.

4.3 Device Specific Errors

Error	Description
-310	<i>System error</i> Indicates that some error, termed “system error” by the device, has occurred.
-311	<i>Memory error</i> Indicates some physical fault in the device’s memory, such as parity error.
-313	<i>Calibration memory lost</i> Indicates that nonvolatile calibration data used by the *CAL? command has been lost.
-314	<i>Save/recall memory lost</i> Indicates that the nonvolatile data saved by the *SAV? command has been lost.
-315	<i>Configuration memory lost</i> Indicates that nonvolatile configuration data saved by the device has been lost.
-321	<i>Out of memory</i> An internal operation needed more memory than was available.
-330	<i>Self-test failed</i> There was an error detected during a self-test. During a self-test the instrument checks the internal hardware for proper function.
-340	<i>Calibration failed</i> The instrument is not or improper calibrated.
-350	<i>Queue overflow</i> A specific code entered into the queue in lieu of the code that caused the error. This code indicates that there is no room in the queue and an error occurred but was not recorded.
-363	<i>Input buffer overrun</i> Software or hardware USB input port buffer overflows with data caused by improper or nonexistent pacing.

4.4 Query Errors

Error	Description
-400	<i>Query error</i> This code indicates that a generic Query Error as defined in IEEE 488.2, 11.5.1.1.7 and 6.3 has occurred.
-410	<i>Query INTERRUPTED</i> Indicates that a condition causing an INTERRUPTED Query error occurred (see IEEE 488.2, 6.3.2.3); for example, a query followed by another query before a response was completely sent.

4.5 Instrument Errors

Error	Description
3	<i>Instrument is overheated</i> The device's internal temperature is too high.
20	<i>Not permitted with LD output on</i> Operation is not allowed while the LD output is switched on.
22	<i>Interlock circuit is open</i> Operation is not allowed while the interlock protection is tripped.
23	<i>Key switch is in locked position</i> Operation is not allowed while the front panel key switch protection is tripped.
24	<i>LD open circuit detected</i> Operation is not allowed while the LD output voltage protection is tripped.
25	<i>LD-ENABLE input is de-asserted</i> Operation is not allowed while the external protection is tripped. This error occurs when the rear panel LD enable input is set to protection mode and has a low level.
26	<i>LD temperature protection is active</i> Operation is not allowed while the temperature protection is tripped. ITC 4000 Series instruments only.
27	<i>Not permitted with photodiode BIAS on</i> Operation is not allowed while the photodiode BIAS is switched on.
28	<i>Not permitted with QCW mode on</i> Operation is not allowed while QCW mode is enabled.
30	<i>Not permitted with TEC output on</i> Operation is not allowed while the TEC output is switched on.
31	<i>Wrong TEC source operating mode</i> Operation is not allowed for the selected TEC operating mode.
32	<i>PID Auto-Tune is currently running</i> Operation not allowed while the PID Auto-Tune procedure is running.
33	<i>PID Auto-Tune value error</i> An PID Auto-Tune procedure was not able to detect proper values.
34	<i>TEC open circuit detected</i> Operation is not allowed while the TEC output voltage protection is tripped.
35	<i>Temperature sensor failure</i> Operation is not allowed while the temperature sensor protection is tripped. E.g. no or wrong sensor connected.
36	<i>TEC cable connection failure</i> Operation is not allowed while the TEC cable connection failure protection is tripped.

5 Appendix

5.1 Instrument Default Settings

The instrument's default settings can be set with the [*SDS](#) ^[12] command. The following table lists the parameters affected by the command.

Table <%TABLENO++%>: Instrument Default Settings

Parameter SCPI command to set parameter	Default Value * <value>
LDC operating mode SOURce[1]:FUNCTION:MODE ^[40]	Constant Current CURR
LDC amplitude modulation SOURce[1]:AM[:STATE] ^[43]	OFF OFF
LDC power feedback SOURce[1]:POWER:ALC:SOURce ^[43]	Photodiode DIOD
LDC power feedback bandwidth SOURce[1]:POWER:ALC:BANDwidth ^[43]	100 Hz 100
LDC switch-on delay OUTPut[1]:DELay ^[29]	2 seconds 2
LDC polarity OUTPut[1]:POLarity ^[30]	cathode grounded CG
LDC current limit SOURce[1]:CURRENT:LIMit[:AMPLitude] ^[40]	e.g. 20 Amperes (*) 20
LDC photodiode current protection SENSe[1][:CURRENT][:DC]:PROTection[:LEVel] ^[46]	e.g. 2 mA (*) 2.0E-03
LDC thermopile/power meter protection SENSe2[:VOLTage][:DC]:PROTection[:LEVel] ^[49]	e.g. 1 Volt (*) 1
LDC output voltage protection (compliance voltage) OUTPut[1]:PROTection:VOLTage[:LEVel] ^[31]	e.g. 1 Volt (*) 1
LDC photodiode polarity INPut[1]:POLarity ^[34]	cathode grounded CG
LDC photodiode current range SENSe[1][:CURRENT][:DC]:RANGe[:UPPer] ^[45]	e.g. 20 mA (*) 0.02
LDC path for photodiode input routing INPut[1]:ROUTe[:TERMinals] ^[36]	LASER OUTPUT D-SUB connector DSUB
LDC photodiode BIAS state INPut[1]:BIAS[:STATE] ^[35]	no BIAS voltage OFF

Parameter	Default Value *
SCPI command to set parameter	<value>
LDC photodiode BIAS voltage	0 Volts
INPut[1]:BIAS:VOLTagE[:DC] ^[35]	0
LDC photodiode responsivity	1 A/W
SENSe[1][:CURRENT][:DC]:CORRection:POWer[:PDIode][:RESPonse] ^[46]	1
LDC thermopile/power meter voltage range	10 Volts
SENSe2[:VOLTagE][:DC]:RANGe[:UPPer] ^[48]	10
LDC path for thermopile/power meter routing	LASER OUTPUT D-SUB connector
INPut2:ROUte[:TERMinals] ^[37]	DSUB
LDC thermopile/power meter responsivity	1 V/W
SENSe2[:VOLTagE][:DC]:CORRection:POWer[:THERmopile][:RESPonse] ^[49]	1
LDC amplitude modulation state	disabled
SOURce[1]:AM[:STATE] ^[43]	OFF
LDC amplitude modulation source	internal
SOURce[1]:AM:SOURce ^[43]	INT
LDC amplitude modulation shape	sinusoidal
SOURce[1]:AM:INTernal:SHAPE ^[43]	SIN
LDC amplitude modulation frequency	10.0 kHz
SOURce[1]:AM:INTernal:FREQuency ^[43]	1.0E+04
LDC amplitude modulation depth	10 %
SOURce[1]:AM:INTernal[:DEPTh] ^[43]	10
LDC enable input function	no function
OUTPut[1]:PROTection:EXTeRnal[:MODE] ^[32]	OFF
LDC internal temperature protection function	no function
OUTPut[1]:PROTection:INTernal[:MODE] ^[31]	OFF
LDC pulsed constant current (QCW) trigger source	internal
TRIGger[:SEQuence]:SOURce ^[51]	INT
LDC pulsed constant current (QCW) mode pulse period	20 ms
SOURce[1]:PULSe:PERiod ^[44]	0.02
LDC pulsed constant current (QCW) mode pulse width	1 ms
SOURce[1]:PULSe:WIDTh ^[44]	1.0E-03
LDC pulsed constant current (QCW) mode duty cycle	5 %
SOURce[1]:PULSe:DCYCLE ^[44]	5
LDC pulsed constant current (QCW) mode hold parameter	pulse width
SOURce[1]:PULSe:HOLD ^[44]	WIDT

Parameter	Default Value *
SCPI command to set parameter	<value>
LDC laser current setpoint	0 Amperes
SOURce[1]:CURRent[:LEVel][:IMMediate] ^[41] [:AMPLitude] ^[41]	0
LDC photodiode current setpoint	0 Amperes
SOURce[1]:POWer[:LEVel]:DIODE[:CURRent] ^[41] [:IMMediate] ^[41]	0
LDC thermopile/power meter voltage setpoint	0 Volts
SOURce[1]:POWer[:LEVel]:PMETER[:VOLTage] ^[41] [:IMMediate] ^[41]	0
LDC power setpoint	0 Watts
SOURce[1]:POWer[:LEVel][:IMMediate] ^[41] [:AMPLitude] ^[41]	0
TED operating mode	const. temperature TEMP
SOURce{[1] 2}:FUNctIon[:MODE] ^[53]	
TED current limit	e.g. 0.1 A (*) 0.1
SOURce{[1] 2}:CURRent:LIMit[:AMPLitude] ^[54]	
TED current setpoint	0 Amperes 0
SOURce{[1] 2}:CURRent[:LEVel][:IMMediate] ^[54] [:AMPLitude] ^[54]	
TED sensor selection	AD590 sensor AD590
SENSe{[1] 3}:TEMPerature:TRANsducer[:TYPE] ^[60]	
TED transducer offset	0°C / 0K 0
SENSe{[1] 3}:TEMPerature:OFFSet ^[61]	
TED temperature protection window	5.0°C 5
SENSe{[1] 3}:TEMPerature:PROtEction ^[62] :WINDow[:AMPLitude] ^[62]	
TED temperature protection window reset delay	1 second 1
SENSe{[1] 3}:TEMPerature:PROtEction:DELay ^[62]	
TED PID control loop - gain value / P-share	1.0 A/K 1
SOURce{[1] 2}:TEMPerature:LCONstants[:GAIN] ^[56]	
TED PID control loop - integral value / I-share	0.1 A*s/K 0.1
SOURce{[1] 2}:TEMPerature:LCONstants:INTegral ^[56]	
TED PID control loop - derivative value / D-share	0.0 A/K*s 0
SOURce{[1] 2}:TEMPerature:LCONstants ^[56] :DERivative ^[56]	
TED PID control loop - period	1.0 s 1
SOURce{[1] 2}:TEMPerature:LCONstants:PERiod ^[56]	

Parameter	Default Value *
SCPI command to set parameter	<value>
TED temperature setpoint limit high	150.0°C
SOURCE{[1] 2}:TEMPERature:LIMit:HIGH ^[55]	150
TED temperature setpoint limit low	-55.0°C
SOURCE{[1] 2}:TEMPERature:LIMit:LOW ^[55]	-55
TED temperature setpoint	25°C
SOURCE{[1] 2}:TEMPERature[:SPOint] ^[55]	25
TED thermistor R ↔ T calculation method	exponential meth.
SENSe{[1] 3}:TEMPERature:THERmistor:METHOD ^[60]	EXP
TED thermistor Steinhart Hart parameter A	1.129241×10 ⁻³
SENSe{[1] 3}:TEMPERature:THERmistor: METHOD[:SHH]:A ^[60]	1.129241E-03
TED thermistor Steinhart Hart parameter B	2.341077×10 ⁻⁴
SENSe{[1] 3}:TEMPERature:THERmistor: METHOD[:SHH]:B ^[60]	2.341077e-04
TED thermistor Steinhart Hart parameter C	8.775468×10 ⁻⁸
SENSe{[1] 3}:TEMPERature:THERmistor: METHOD[:SHH]:C ^[60]	8.775468E-08
TED thermistor exp. method parameter R ₀	10.0 kΩ
SENSe{[1] 3}:TEMPERature:THERmistor: METHOD:EXPOntial:R0 ^[60]	1.0E+04
TED thermistor exp. method parameter T ₀	25.0°C
SENSe{[1] 3}:TEMPERature:THERmistor: METHOD:EXPOntial:T0 ^[60]	25
TED thermistor exp. method parameter B-value	3575.0
SENSe{[1] 3}:TEMPERature:THERmistor: METHOD:EXPOntial:BETA ^[60]	3575
Display brightness	100 %
DISPlay:BRIGHTness ^[75]	100
Display contrast	45 %
DISPlay:CONTRast ^[75]	45
Acoustic feedback	enabled
SYSTem:BEEPer[:IMMediate] ^[16]	ON
Temperature unit	degree Celsius
UNIT:TEMPERature ^[79]	C
Line frequency setting	autom. detection
SYSTem:LFRequency ^[17]	AUTO

(*) The default values stated here are typical values, they may vary from model to model. For more information on default values please see the user's manual provided with the

instrument.

5.2 Copyright and Exclusion of Reliability

Thorlabs GmbH has taken every possible care in preparing this Operation Manual. We however assume no liability for the content, completeness or quality of the information contained therein. The content of this manual is regularly updated and adapted to reflect the current status of the software. We furthermore do not guarantee that this product will function without errors, even if the stated specifications are adhered to.

Under no circumstances can we guarantee that a particular objective can be achieved with the purchase of this product.

Insofar as permitted under statutory regulations, we assume no liability for direct damage, indirect damage or damages suffered by third parties resulting from the purchase of this product. In no event shall any liability exceed the purchase price of the product.

Please note that the content of this User Manual is neither part of any previous or existing agreement, promise, representation or legal relationship, nor an alteration or amendment thereof. All obligations of *Thorlabs GmbH* result from the respective contract of sale, which also includes the complete and exclusively applicable warranty regulations. These contractual warranty regulations are neither extended nor limited by the information contained in this User Manual. Should you require further information on this product, or encounter specific problems that are not discussed in sufficient detail in the User Manual, please contact your local *Thorlabs GmbH* dealer or system installer.

All rights reserved. This manual may not be reproduced, transmitted or translated to another language, either as a whole or in parts, without the prior written permission of *Thorlabs GmbH*.

Copyright © Thorlabs GmbH 2015. All rights reserved.

5.3 Thorlabs Worldwide Contacts

For technical support or sales inquiries, please visit us at www.thorlabs.com/contact for our most up-to-date contact information.



USA, Canada, and South America

Thorlabs, Inc.
sales@thorlabs.com
techsupport@thorlabs.com

Europe

Thorlabs GmbH
europe@thorlabs.com

France

Thorlabs SAS
sales.fr@thorlabs.com

Japan

Thorlabs Japan, Inc.
sales@thorlabs.jp

UK and Ireland

Thorlabs Ltd.
sales.uk@thorlabs.com
techsupport.uk@thorlabs.com

Scandinavia

Thorlabs Sweden AB
scandinavia@thorlabs.com

Brazil

Thorlabs Vendas de Fotônicos Ltda.
brasil@thorlabs.com

China

Thorlabs China
chinasales@thorlabs.com

Index

*			
*IDN?	11	CURRent	26, 41
*OPC	12	CURRent:LIMit	40
*OPC?	12	CURRent[:LEVel]	41
*RCL	12	D	
*RST	11	Device Specific Errors	84
*SAV	12	Digital I/O Configuration	77
*SDS	12, 86	Digital I/O Input Commands	77
*TST?	12	Digital I/O Output Commands	76
*WAI	13	discrete parameter	9
		DISP:CONT	75
A		DISPlay	75
ABORt	22	Display Subsystem Commands	75
ALC	41	DISPlay:BRIGHTness	75
AMPLitude	40, 41	E	
amplitude modulation source(s)	43	enable register	68
ASCII	9, 29	ERRor	16
Auto-PID	56	event register	68
Auxiliary Register Group	69	Execution Errors	83
		F	
B		FETCh	24
BANDwidth	41	FILTer	30
BEEPer	16	filter register	68
BIAS	35	FUNCTion	40
BIAS:VOLTage	35	FUNCTion:MODE	40
Boolean	9	I	
Boolean parameters	9	IEEE488.2 commands	10
brightness	75	INITiate	24
		INPut	34
C		INPut2	37
calculation method for a thermistor	60	INPut2:ROUte	37
Calibration Commands	80	Instrument Errors	85
CALibration:STRing	80	instrument's default settings	86
colon	7	INTLock	33
comma	7	Introduction to the SCPI Language	6
condition register	68	ITC4000	5, 29
CONFigure	22	K	
contrast	75	KEYLock	33
CORR	46		
CORR:POW	46, 49		

		PID temperature control loop	56
L		POLarity	30, 34
LD output power	41	POW	46
LD output protection voltage	31	POW:PROT	47, 50
LD source current setpoint	41	POW:RANG	47, 50
LD source limit current	40	POWer	26, 41
LDC4000	5, 29	power limit	47
LD-ENABLE input mode	32	power limit value	50
LPASs	30	power range	47
		power source feedback loop input	43
M		PROT	46, 49
MEASure	26	PROTection	31, 33, 47
Measurement Commands	18	Protection Queries	33, 64
Measurement Register Group	70	PROTection:EXTeRnal	32
MEM:STAT:NAME	74	PROTection:INTeRnal	31
MEMory	74	Q	
Memory Subsystem Commands	74	QCW mode triggering	51
MEMory:NSTates	74	QCW Pulse Setting	44
		QCW pulse trigger	51
N		Query Errors	84
name of state memory	74	Questionable Register Group	71
noise reduction fi	30	R	
number of state memories	74	RANG	45, 47, 48
numeric parameters	8	READ	26
O		ROUTe	36
Operation Registers Group	72	ROUTe[:TERMiNals]	37
OTEMP	33	S	
OUTP	63	SCALar	26
OUTP:PROT:CABL	64	SCPI	6, 8
OUTPut	28	SCPI command to set parameter	86
Output State	29	SCPI Error Messages	81
OUTPut[1]:PROTection:OTE Mp	64	SCPI Status System	73
OUTPut[1]:PROTection:TRA N	64	semicolon	7
P		SENS:CORR:POW	46
PD input	34	SENS:POW:PROT	47, 50
PD input BIAS voltage	35	SENS:POW:RANG	47
PD input routing	36	SENS:PROT	46
photodiode (PD) sense	45	SENS:RANG	45
photodiode current limit	46	SENS:TEMP	60
photodiode responsivity	46	SENS:TEMP:OFFS	61

SENS:TEMP:PROT:WIND	62	TEC output	63
SENS:TEMP:THER:METH	60	TEC source	52
SENS2:CORR:POW	49	TEC source function	53
SENS2:POW:RANG	50	TED4000	5, 53, 54, 55, 56, 62
SENS2:PROT	49	temperature	26, 62
SENS2:RANG	48	temperature offset	61
SENSe[1]	45	temperature protection mode	31
SENSe{2 3 4}:DIGital:DATA	78	temperature sensing	58
SENSe2	47	temperature sensor	60
SHAPE	40	temperature setpoint	55
SOUR	43, 51	temperature setpoint range	55
SOUR:AM	43	temperature window	62
SOUR:AM:INT:FREQ	43	TEMPerature:ATUNe	56
SOUR:CURR	54	thermopile/power meter input	37
SOUR:CURR:LIM	54	thermopile/power meter input routing	37
SOUR:FUNC	53	thermopile/power meter power range	50
SOUR:PULS:DCYC	44	thermopile/power meter responsivity	49
SOUR:PULS:HOLD	44	thermopile/power meter sense	47
SOUR:PULS:PER	44	thermopile/power meter	49
SOUR:TEMP	55	voltage limit	
SOUR:TEMP:LCON	56	TRIG:SOUR	51
SOUR:TEMP:LIM:	55	TRIGger	51
SOUR2:TEMP	55	TRIPped	40
SOUR3:DIG	76	TSEnSor	26
SOUR3:DIG:DATA	76		
SOUR3:DIG:ENAB	77	U	
SOURce[1]	38	Unit Commands	79
SOURce[1]:POWer	41	UNIT subsystem	79
SOURce[1]:POWer:ALC:SO URce	43	UNIT:TEMP	79
SOURce{2 3}	76		
state of the PD input BIAS	35	V	
Status Preset Command	73	VERSIon	17
Status Subsystem	65	VOLTage	26
STATus:AUXiliary	69		
STATus:MEASurement	70	W	
STATus:OPERation	72	WIDT	44
STATus:PRESet	73		
STATus:QUEStionable	71		
switch-on delay	29		
SYSTEM	16		
System Subsystem Commands	16		

T

TEC current setpoint	54
TEC element limit current	54