

Computer Interfacing Manual

Table of Contents

INTRODUCTION	3
THE RS232 INTERFACE	3
CABLE CONNECTIONS	3
INTERFACE SETTINGS	3
THE USB INTERFACE	4
USING THE ARROYO INSTRUMENTS USB INTERFACE	4
LOADING THE SOFTWARE DRIVERS	4
CABLE CONNECTIONS AND INTERFACE SETTINGS	4
USING COMMANDS	4
IEE-488.2 COMMANDS	5
DEVICE-INDEPENDENT COMMANDS	5
DEVICE-DEPENDENT COMMANDS	5
LASERSOURCE COMMAND LIST	5
TECSOURCE COMMAND LIST	6
COMMAND SYNTAX AND CONCATENATION	6
COMMAND PATHS	6
NUMERIC SUBSTITUTIONS	7
EVENT AND CONDITION REPORTING	7
INSTRUMENT STATUS REPORTING STRUCTURE	8
ALPHABETICAL LIST OF COMMANDS	9
ERROR MESSAGES	54

Introduction

The *Computer Interfacing Manual* provides a complete summary of all commands supported by the Arroyo Instruments line of products. While in remote control mode, the computer interface allows full operation of the instrument, plus advanced features only available via the computer interface.

You will find in reviewing the command set that it is largely compatible with both ILX and Newport laser diode drivers and temperature controllers. In fact, for most applications, you can place an Arroyo Instruments controller in place of a Newport or ILX instrument and the software application would work with little or no changes.

The RS232 Interface

The RS232 interface provides a common, easy-to-use interface for controlling the instrument. With baud rates up to 115k baud, high-speed control and measurement is possible.

Cable Connections

The RS232 interface is a male DB9 pinned identically to a standard PC RS232 port, so a female/female cable in NULL modem configuration (transmit and receive pins swapped) is required. The full pin-out is described below:

Pin	Description
2	Receive
3	Transmit
5	Ground
Shell	Earth ground

RS232 Connector (DB-9 Male)

Depending on if you are connecting to a DB9 or DB25 on the PC, follow the pin-to-pin assignments in the table below.

Instrument	PC DB9	PC DB25
2	3	2
3	2	3
5	5	7

Instrument to PC pin assignment

Interface Settings

Baud Rate	The instrument will operate at 300, 1200, 2400, 4800, 9600, 19200, 38400, 57600, and 115200 baud rates. The baud rate can be set through the instrument's menu, and must match the setting on the PC
Parity	None
Data Bits	8
Stop Bits	1
Flow Control	None

The USB Interface

Because of its speed, expandability, commonality, the USB interface has become the interface of choice for newer PC-connected devices. Unlike GPIB, USB uses inexpensive cables and allows up to 127 devices to be connected to a single USB master.

Using the Arroyo Instruments USB Interface

To keep complexity to a minimum, once you have installed the USB drivers, the instrument will appear as a virtual serial port that you can use just like a normal serial port. In this way, you can communicate to the instrument without requiring special software modifications to your existing applications.

Loading the Software Drivers

Unlike the RS232 interface, using the USB interface requires software drivers (provided on the CD with your instrument) to be loaded onto your PC. Follow the instructions in your instrument manual for installing the USB drivers.

Once the drivers are installed, to determine the COM port number, go to **Control Panel** and select **System**. Once the **System Properties** dialog appears, choose the **Hardware** tab then click on the **Device Manager** button. When the **Device Manager** appears, click on the plus sign to the left of **Ports**. The port identified as an **USB Serial Port** is the instrument. In the event you have multiple instruments plugged in simultaneously, you will need to experiment to see which instrument was assigned to which port. For example, you could change the set point when the output was off to see which unit's set point changed.

Once a COM port has been assigned to an instrument, it will continue to use the same COM port unless it is reassigned through the Control Panel.

Cable Connections and Interface Settings

The instrument USB interface is a full size USB Type B socket. You may use any USB 1.1 or 2.0 certified cable.

Baud Rate	38400
Parity	None
Data Bits	8
Stop Bits	1
Flow Control	None

Using Commands

While the instrument does not have a GPIB interface, the command structure and parsing were developed around the IEEE-488.2 standard. Most IEEE-488.2 commands are supported, and the command/response format is consistent with the standard.

Commands are grouped into two major categories: device-independent and device-dependent commands. Device-independent commands include all the IEEE-488.2 supported commands plus other commands that are universal across all instruments. Device-dependent commands are specific to a class of instruments, such as TEC commands for the TECSource. Not all instruments support every command. Review the command for instrument-specific differences.

IEE-488.2 Commands

The following device-independent commands are supported by most instruments:

*CLS	*IDN?	*PSC	*SRE
*ESE	*IST?	*PSC?	*SRE?
*ESE?	*PRE	*RCL	*STB?
*ESR?	*PRE?	*RST	*TST?
*WAI			

Device-Independent Commands

The following device-independent commands are supported by most instruments:

BEEP	ERRors?	ONDELAY?	TERM?
BEEP?	ERRSTR?	RADix	TERMINAL
BRIGHT	LOCAL	RADix?	TERMINAL?
BRIGHT?	MESsage	REMERR	TIME?
DELAY	MESsage?	REMERR?	TIMER?
EQUIPment?	ONDELAY	TERM	

Device-Dependent Commands

Device-dependent commands are specific to a class of instruments, such as the TECSource.

Many of the device dependent commands have two commands that do the same thing. For example, LAS:I? and LAS:LDI? perform the same operation. The “I”, “IPD”, and “Ppd” forms of commands are considered obsolete but included for compatibility reasons.

LaserSource Command List

LASer:CALMD (CALPD)	LASer:LDV	LASer:MODE:MDI (IPD)
LASer:CALMD? (CALPD?)	LASer:LDV?	LASer:MODE:MDP (Ppd)
LASer:COND?	LASer:LIMit:LDI (I)	LASer:OUTput
LASer:DEC	LASer:LIMit:LDI? (I?)	LASer:OUTput?
LASer:DC	LASer:LIMit:LDV	LASer:PDBias
LASer:DC?	LASer:LIMit:LDV?	LASer:PDBias?
LASer:DELAYIN	LASer:LIMit:MDI (IPD)	LASer:PW?
LASer:DELAYIN?	LASer:LIMit:MDI? (IPD?)	LASer:PWF
LASer:DELAYOUT	LASer:LIMit:MDP (Ppd)	LASer:PWP
LASer:DELAYOUT?	LASer:LIMit:MDP? (Ppd?)	LASer:QCWHOLD
LASer:DISplay	LASer:MDI (IPD)	LASer:QCWHOLD?
LASer:DISplay?	LASer:MDI? (IPD?)	LASer:QCWCOUNT
LASer:ENABLE:COND	LASer:MDP (Ppd)	LASer:QCWCOUNT?
LASer:ENABLE:COND?	LASer:MDP? (Ppd?)	LASer:SET:LDI? (I?)
LASer:ENABLE:EVENT	LASer:MODE?	LASer:SET:LDV?
LASer:ENABLE:EVENT?	LASer:MODE:CW	LASer:SET:MDI? (IPD?)
LASer:ENABLE:OUTOFF	LASer:MODE:ICW	LASer:SET:MDP? (Ppd?)
LASer:ENABLE:OUTOFF?	LASer:MODE:ILBW	LASer:STB?
LASer:EVENT?	LASer:MODE:I	LASer:STEP
LASer:F	LASer:MODE:IHBW	LASer:STEP?
LASer:F?	LASer:MODE:PULSE	LASer:TOLerance
LASer:INC	LASer:MODE:BURST	LASer:TOLerance?
LASer:LDI (I)	LASer:MODE:TRIG	LASer:VSENSE
LASer:LDI? (I?)	LASer:MODE:LDV	LASer:VSENSE?

TECSource Command List

TEC:AUTOTUNE	TEC:FAN?	TEC:OUTput
TEC:AUTOTUNE?	TEC:GAIN	TEC:OUTput?
TEC:AUTOTUNESTATE?	TEC:GAIN?	TEC:PID
TEC:CABLER	TEC:HEATCOOL	TEC:PID?
TEC:CABLER?	TEC:HEATCOOL?	TEC:R
TEC:CABLETYPE?	TEC:INC	TEC:R?
TEC:COND?	TEC:ITE	TEC:SENsor
TEC:CONST	TEC:ITE?	TEC:SENsor?
TEC:CONST?	TEC:LIMit:ITE	TEC:SET:ITE?
TEC:DEC	TEC:LIMit:ITE?	TEC:SET:R?
TEC:DISplay	TEC:LIMit:RHI	TEC:SET:T?
TEC:DISplay?	TEC:LIMit:RHI?	TEC:STB?
TEC:CABLER	TEC:LIMit:RLO	TEC:STEP
TEC:CABLER?	TEC:LIMit:RLO?	TEC:STEP?
TEC:ENABLE:COND	TEC:LIMit:THI	TEC:T
TEC:ENABLE:COND?	TEC:LIMit:THI?	TEC:T?
TEC:ENABLE:EVEnt	TEC:LIMit:TLO	TEC:TOLerance
TEC:ENABLE:EVEnt?	TEC:LIMit:TLO?	TEC:TOLerance?
TEC:ENABLE:OUTOFF	TEC:MODE?	TEC:V?
TEC:ENABLE:OUTOFF?	TEC:MODE:ITE	
TEC:EVEnt?	TEC:MODE:R	
TEC:FAN	TEC:MODE:T	

Command Syntax and Concatenation

All commands use a common syntax for constructing the command and parameter passing. Commands are case-insensitive, and some commands have optional characters, which are denoted as lower-case characters in the command definition. For example, `LASer:EVEnt?` can be shortened to `LAS:EVE?` This allows for command abbreviation and therefore reduced communication times.

The input buffer of the instrument allows for up to 128 characters to be received before a command overflow error occurs. If a command overflow does occur, the entire buffer is discarded and an E-102 (Message too long) error will occur.

If a command requires one or more parameters, place a space between the command and the first parameter, then use a comma to separate additional parameters. For example, the laser tolerance command might look like:

```
LASER:TOLERANCE 10,25
```

It is also possible to concatenate multiple commands together as a single communication to the instrument. By using a semicolon, multiple commands can be sent to the instrument at once, but command processing will not start until all commands and the command terminator have been received. For example, to set the laser set point to 10mA and turn the output on, the command would look like:

```
LASER:LDI 10; LASER:OUTPUT 1
```

Command Paths

The command set is structured in a tree-like organization, where commands for a common device or function are grouped together. For example, all commands specific to the LaserSource start with a "LASer:" prefix. This is called the command path.

When issuing multiple concatenated commands to the instrument, it “remembers” the command path of the previous command, and allows you to omit the common path from the second command. Using the example above where the set point was changed to 10mA and the output turned on, it could also be shortened by omitting the second “LASER:” portion because when the OUTPUT command is processed, the command parser “remembers” it’s position in the command tree and starts looking for the command at that level. The command could therefore be shortened to this:

```
LASER:LDI 10;OUTPUT 1
```

Removing the optional characters from the commands, you can further shorten the command to:

```
LAS:LDI 10;OUT 1
```

In the case where the same command exists at the last command level and in one or more levels above, you can start the command with a colon (':') to force the command parser to start at the root. For example, take the following command:

```
LASER:MODE:LDV;LDV 0.5
```

At first glance, it appears to set the laser into voltage control mode and then set the set point to 0.5V. However, after processing the first command, the parser is at the “LASER:MODE:” level in the command tree and the second “LDV” command will match the “LASER:MODE:LDV” command, basically changing modes twice. To force the parser to interpret the command the way we intended, compose the second command after the semicolon like this:

```
LASER:MODE:LDV; :LASER:LDV 0.5
```

Numeric Substitutions

For command readability, you can also substitute alternate values for zero (0) and one (1). “0” can be replaced by “OFF”, “NEW”, or “FALSE”. “1” can be replaced by “ON”, “OLD”, or “TRUE”.

Event and Condition Reporting

In order to allow for quick summary reporting of device status, the instruments contain a powerful status reporting structure, which can be configured to report status at several levels of the instrument’s operation. Using the various enable registers, a summary of the instrument’s operation is made into the status byte register, allowing the control application to query only the status byte register, and if no pertinent events or conditions are being reported, no other status checking is needed.

In some cases, condition and event registers may appear to report the same thing, but it is how the registers work that make them unique from each other. A condition register will report the status of the instrument at the moment in time when the condition register is queried. For fast conditions (such as laser over-voltage), it’s not possible to detect the condition no matter how fast you query the instrument. This is where event registers come in: they “remember” their state until you read them. For example, if a laser goes into current limit, then out of current limit, its condition register will indicate current limit only while it is in that state. The event register, however, will set its current limit bit, and that bit will remain set until the event register is read. All event registers, once read, will automatically reset themselves to zero.

For example, if you wanted to monitor if the laser was in current or power limit, if the output turned off, or if there were any error in the error queue, you could enable reporting of current or power limit conditions in the LASer:COND register, and enable the output status change event in the LASer:EVEnt register. By

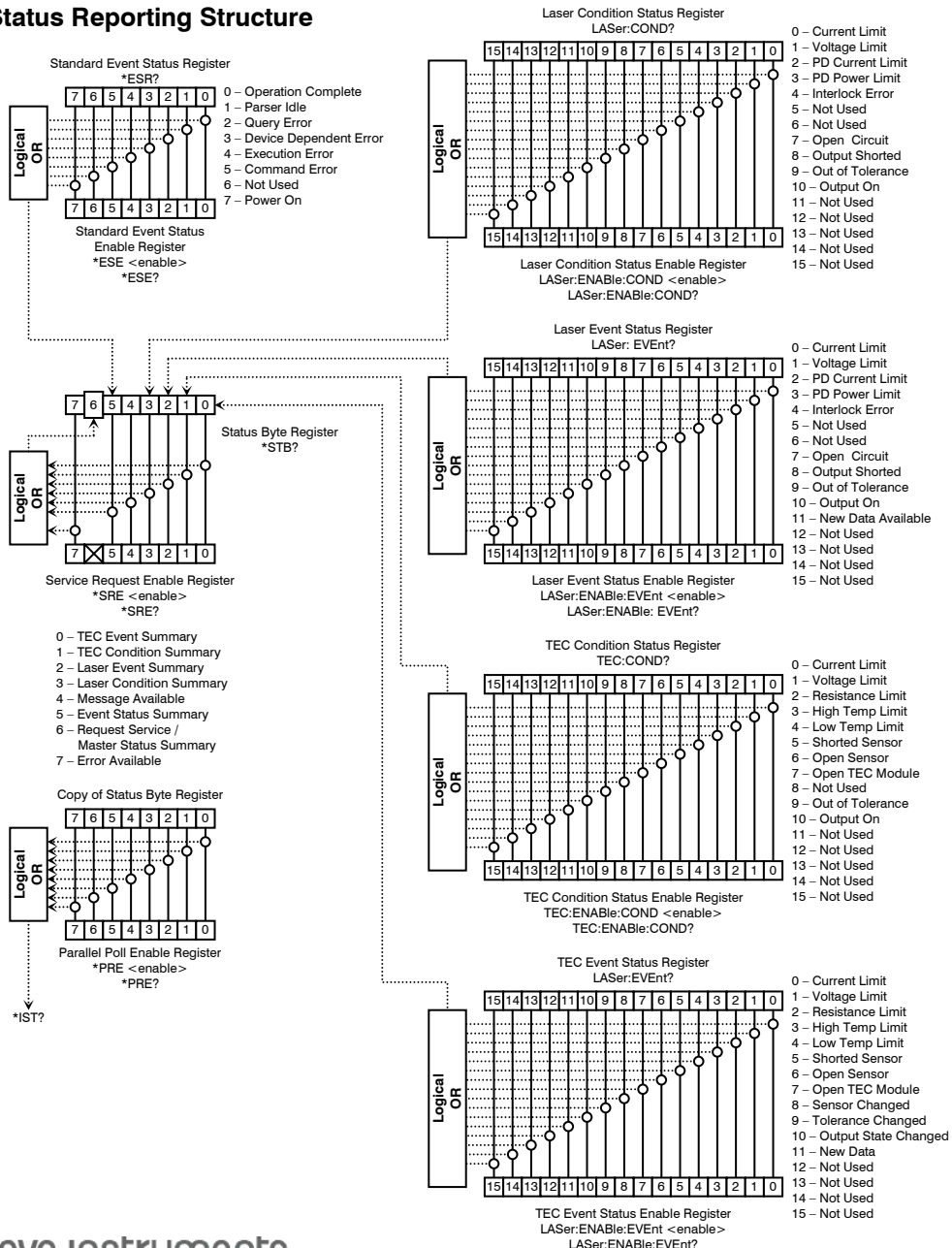
querying the status byte register (*STB?), no further action is necessary if the laser event, laser condition, and error available bits were all zero. If you did not use the status reporting system, you would need to issue at least three commands to do the same thing.

At first glance, the reporting structure pictured on the next page can seem complex, but it is actually quite simple. For each register, there is a corresponding enable register. For example, the laser condition register (LASer:COND) has a laser condition enable register (LASer:ENABLE:COND). When the same bits from each register are on (or 1 or true), then the bit in the status byte register is turned on (in this case, bit 3, the laser condition summary bit).

By reviewing your application needs and the reporting capabilities of the system, you can determine what conditions or event you want to enable and appropriately set each of the enable registers.

In USB and RS232 applications, the Service Request Enable Register, the Parallel Poll Enable Register, and the IST bit have little value because the Status Byte Register will contain more detail, but these registers have been implemented for greater compatibility with existing software applications.

Instrument Status Reporting Structure



Alphabetical List of Commands

*CLS

Synopsis Clear status registers command

Syntax *CLS

Details Clears the standard event status register, all event registers, and the error queue.

See Also ESR?, ERR?, *IST?, *STB?

*ESE

Synopsis Set the Event Status Register Enable

Syntax *ESE *mask*

Details See the ESR? query for a definition of the individual bits with the *mask* parameter. The *mask* is logically ANDed with the ESR register, and if any of the resulting bits are high, bit 5 of the STB register (Event Status Summary) is set.

See Also *ESE?, ESR?, *STB?

*ESE?

Synopsis Query the Event Status Register Enable

Syntax *ESE?

Details Returns the value of the Event Status Register Enable. See the ESR? query for a definition of the individual bits with the response value.

See Also *ESE, ESR?, *STB?

*ESR?

Synopsis Query the Event Status Register

Syntax *ESR?

Details Returns the value of the Event Status Register. As with all event registers, after the value has been read, the register will be set to zero.

Response	Bit	Value	Description
<i>ESR</i>	0	1	Operation complete
	1	2	Parser idle: the parser buffer is empty
	2	4	Query error (errors 300 through 399)
	3	8	Device dependent error (errors 400 through 599)
	4	16	Execution error (errors 200 through 299)
	5	32	Command error (errors 100 through 199)
	6	64	Not Used
	7	128	Power on: the unit has been powered on since the last time this register was read or cleared.

Operation complete indicates when the controller has completed all pending operations and the outputs are stable.

See Also *ESE, ESE?, *STB?

*IDN?

Synopsis Query the instrument identification

Syntax *IDN?

Details Returns the identification string for the instrument in the following format:

Arroyo Model SN Ver Build

Response	Description
<i>Model</i>	The model number of the product, such as '4205'.
<i>SN</i>	The serial number.
<i>Ver</i>	The firmware version.
<i>Build</i>	Internal build number.

*IST?

Synopsis Query Individual Status

Syntax *IST?

Details Returns the value of the 'ist' bit within the reporting structure.

Response	Value	Description
<i>ist</i>	0	ist false
	1	ist true

See Also *PRE, *STB

*PRE

Synopsis Set the Parallel Poll Enable Register

Syntax *PRE *mask*

Details See the STB? query for a definition of the individual bits with the *mask* parameter. The *mask* is logically ANDed with the Status Byte Register, and if any of the resulting bits are high, the 'ist' bit is set.

See Also *PRE?, *STB?

*PRE?

Synopsis Query the Parallel Poll Enable Register

Syntax *PRE?

Details Returns the value of the Parallel Poll Enable register. See the STB? query for a definition of the individual bits with response value.

See Also *PRE, *STB?

*PSC

Synopsis Set the Power-On Status Clear

Syntax *PSC *enable*

Details Enables or disables the power-on clearing of event and condition status enable registers.

Argument	Value	Description
<i>enable</i>	1	All event and condition status enable registers are cleared at power-up.
	0	All event and condition status enable registers are restored their last power-off state.

See Also *PSC?

*PSC?

Synopsis Query the Power-n Status Clear

Syntax *PSC?

Details Returns the value of the Power-On Status Clear register. See the *PSC command for a definition of the response value.

See Also *PSC

*RCL

Synopsis Recall command

Syntax *RCL 0

Details The *RCL command is used to restore the unit to saved configuration. However, only '*RCL 0' is support, and behaves identically to a *RST command.

See Also *RST

*RST

Synopsis Reset command

Syntax *RST

Details Resets the instrument to factory defaults, and the output is shut off. The unit remains in remote mode.

See Also *RCL

*SRE

Synopsis Set the Service Request Enable

Syntax *SRE *mask*

Details The *SRE command sets the Service Request Enable Register bits, which control the Master Status Summary bit in the Status Byte Register.

Argument	Bit	Value	Description
<i>mask</i>	0	1	TEC Event Summary
	1	2	TEC Condition Summary
	2	4	Laser Event Summary
	3	8	Laser Condition Summary
	4	16	Message Available
	5	32	Event Status Summary
	6	64	Request Service / Master Status Summary
	7	128	Error Message Available

See Also *SRE?

*SRE?

Synopsis Query the Service Request Enable

Syntax *SRE?

Details Returns the value of the Service Request Enable register. See the *SRE command for a definition of the response value.

See Also *SRE

*STB?

Synopsis Query the Status Byte Register

Syntax *STB?

Details Returns the value of the Status Byte Register.

Response	Bit	Value	Description
<i>STB</i>	0	1	TEC Event Summary
	1	2	TEC Condition Summary
	2	4	Laser Event Summary
	3	8	Laser Condition Summary
	4	16	Message Available
	5	32	Event Status Summary
	6	64	Request Service / Master Status Summary
	7	128	Error Message Available

See Also *SRE, *PRE

*WAI

Synopsis Wait for Operation Complete command

Syntax *WAI

Details The *WAI command will pause command processing until the Operation Complete flag is true.

BEEP

Synopsis Set the beep enable

Syntax BEEP [*enable*]

Details Causes the instrument to beep, or enables or disabled the beep sound for error messages and other events that generate an audible response. If *enable* is omitted, it is the equivalent of doing a 'BEEP 2'.

Argument	Value	Description
<i>enable</i>	0	Disable the beep sound
	1	Enable the beep sound
	2	Generate one beep

See Also BEEP?

BEEP?

Synopsis Query the beep enable

Syntax BEEP?

Details Returns the value of the Beep register.

Response	Value	Description
<i>enable</i>	0	Disable the beep sound
	1	Enable the beep sound

See Also BEEP

BRIGHT

Synopsis Set the display brightness

Syntax BRIGHT *brightness*

Details Sets the display brightness level, from 0 to 100 percent.

Argument	Value	Description
<i>brightness</i>	0 to 100	Brightness level of the display

On some displays, such as VFDs, there may only be 4 or 8 brightness levels, and the *brightness* parameter will be used to select an appropriate level.

See Also BRIGHT?

BRIGHT?

Synopsis Query the display brightness

Syntax BRIGHT?

Details Returns the value of the display brightness. See the BRIGHT command for more details.

See Also BRIGHT

DELAY

Synopsis Causes a delay in command processing

Syntax DELAY *time*

Details Causes command processing to be delayed for the specified number of milliseconds.

Argument	Value	Description
<i>time</i>	1 to 30000	Delay, in milliseconds

EQUIPment?

Synopsis Query for installed equipment

Syntax EQUIPment?

Details The EQUIPment? query returns a comma-separated list of installed modules, which is model number from each module. For modular controllers, the first parameter is the model number of the chassis, followed by the model numbers of each module. For single channel instruments, it returns the model number of the instrument.

ERRors?

Synopsis Query for errors

Syntax ERRors?

Details Returns a comma-limited list of error codes. If no error has occurred, a 0 is returned.

A typical response might look like:

```
201,124
```

See Also ERRSTR?

ERRSTR?

Synopsis Query for errors with string descriptions

Syntax ERRSTR?

Details Similar to the ERR? query, but a string description is included with the error code. A typical response might look like:

201, "Out of range", 124, "Data mismatch"

See Also ERR?

LASer:CALMD (CALPD)

Synopsis Set photodiode optical response

Syntax LASer:CALMD *PDresp*

Details Set the photodiode optical response value for converting photodiode current into optical power. This is the PD Response value from the instrument menu.

Argument	Value	Description
<i>PDresp</i>	0.0001 to 1000	Sensitivity in $\mu\text{A}/\text{mW}$

If *PDresp* is zero, then power mode will be unavailable, and the instrument will display photodiode current. If *PDresp* is non-zero, then power mode will be allowed, and the instrument will display optical power.

See Also LASer:CALMD?

LASer:CALMD? (CALPD?)

Synopsis Query the photodiode optical response

Syntax LASer:CALMD? (CALPD?)

Details Returns the value of the photodiode optical response. See the LASer:CALMD command for more details.

See Also LASer:CALMD

LASer:CAL:CANCEL

Synopsis Cancel calibration in progress

Syntax LASer:CAL:CANCEL

Details Cancels any calibration in progress.

LASer:CAL:LDX, LDX?, LVX, LVX?, MDX, MDX?

Synopsis Laser calibration functions

Details See the calibration section for details on how to use these commands.

LASer:COND?

Synopsis Query laser condition

Syntax LASer:COND?

Details Returns the laser condition register.

Response	Bit	Value	Description
<i>conditions</i>	0	1	Current limit
	1	2	Voltage limit
	2	4	Photodiode current limit
	3	8	Photodiode power limit
	4	16	Interlock disabled
	5	32	Unused
	6	64	Unused
	7	128	Laser open circuit
	8	256	Laser shorted
	9	512	Out of tolerance
	10	1024	Output on
	11	2048	Unused
	12	4096	Unused
	13	8192	Unused
	14	16384	Unused
15	32768	Unused	

See Also LASer:ENABLE:COND, *STB?

LASer:DC

Synopsis Set the duty cycle in QCW mode

Syntax LASer:DC *dutycycle*

Details This changes the duty cycle percentage when in QCW mode. A new frequency value will be computed based on the duty cycle and pulse width (the pulse width is not changed). If the resulting frequency is outside the allowable range, an E-201 error will be generated and the command ignored.

Argument	Description
<i>dutycycle</i>	Duty cycle, in percent

See Also LASer:DC?, LASer:F, LASer:PWF, LASer:PWP

LASer:DC?

Synopsis Query the QCW duty cycle

Syntax LASer:DC?

Details Returns the value of the duty cycle. See LASer:DC for a definition of the *dutycycle* response value.

See Also LASer:DC

LASer:DEC

Synopsis Decrement the laser set point

Syntax LASer:DEC *steps* [, *time*]

Details The LASer:DEC command uses the step size defined with the LASer:STEP command to decrement the laser set point. If the *time* parameter is omitted, then the set point is immediately decremented *steps* times the step size. If the *time* parameter is included,

then the instrument will still decrement *steps* times, but will pause at each step for *time* milliseconds.

Argument	Value	Description
<i>steps</i>	1 to 65000	Number of steps to decrement
<i>time</i>	0 to 65000	Number of milliseconds to pause between each step

See Also LASer:INC, LASer:STEP

LASer:DELAYIN

Synopsis Set the pulse delay, in seconds, between the trigger input and output pulse

Syntax LASer:DELAYIN *delay*

Details Sets the delay from the rising edge of TRIGIN to the generation of the output pulse in QCW mode.

Argument	Value	Description
<i>delay</i>	0.000015 to 1	Delay, in seconds

See Also LASer:DELAYIN?, LASer:DELAYOUT

LASer:DELAYIN?

Synopsis Query the trigger input pulse delay

Syntax LASer:DELAYIN?

Details Returns the value of the trigger input delay. See LASer:DELAYIN for a definition of the *delay* response value.

See Also LASer:DELAYIN

LASer:DELAYOUT

Synopsis Set the trigger output delay, in seconds, between the output pulse and the trigger output

Syntax LASer:DELAYOUT *delay*

Details Sets the delay from the rising edge of the pulse to the rising edge of TRIGOUT in QCW mode.

Argument	Value	Description
<i>delay</i>	0.000000 to 1	Delay, in seconds

See Also LASer:DELAYOUT?, LASer:DELAYIN

LASer:DELAYOUT?

Synopsis Query the trigger output delay

Syntax LASer:DELAYOUT?

Details Returns the value of the trigger output delay. See LASer:DELAYOUT for a definition of the *delay* response value.

See Also LASer:DELAYOUT

LASer:DISPlay

Synopsis Set laser display enable

Syntax LASer:DISPlay *enable*

Details The LASer:DISPlay command can be used to completely lock out local operation of the instrument and display “Local Lockout, Display Disabled” instead of the normal display.

Response	Value	Description
<i>enable</i>	0	Disables the display and front panel
	1	Enables the display and front panel

Once the display is disabled, the front panel is completely locked out. The only way to restore functionality to the front panel is to issue a “LASer:DISPlay 1” or cycle power on the unit.

See Also LASer:DISPlay?

LASer:DISPlay?

Synopsis Query the display enable state

Syntax LASer:DISPlay?

Details Returns the value of the laser display enable state. See LASer:DISPlay for a definition of the *enable* response value.

See Also LASer:DISPlay

LASer:ENABLE:COND

Synopsis Set Laser Condition Enable register

Syntax LASer:ENABLE:COND *conditions*

Details Enables reporting of selected conditions to the Status Byte Register. See the LASer:COND command for a definition of the *conditions* parameter.

See Also LASer:ENABLE:COND?, LASer:COND?

LASer:ENABLE:COND?

Synopsis Query Laser Condition Enable register

Syntax LASer:ENABLE:COND?

Details Returns the value of the Laser Condition Enable register. See the LASer:COND command for a definition of the *conditions* response.

See Also LASer:ENABLE:COND, LASer:COND?

LASer:ENABLE:EVENT**Synopsis** Set Laser Event Enable register**Syntax** LASer:ENABLE:EVENT *events***Details** Enables reporting of selected events to the Status Byte Register. See the LASer:EVENT command for a definition of the *events* parameter.**See Also** LASer:ENABLE:EVENT?, LASer:EVENT?

LASer:ENABLE:EVENT?**Synopsis** Query Laser Event Enable register**Syntax** LASer:ENABLE:EVENT?**Details** Returns the value of the Laser Event Enable register. See the LASer:EVENT command for a definition of the *events* response.**See Also** LASer:ENABLE:EVENT, LASer:EVENT?

LASer:ENABLE:OUTOFF**Synopsis** Set the Output Off Enable register**Syntax** LASer:ENABLE:OUTOFF *outoff***Details** The Output Off register controls what conditions will cause the laser output to be turned off. Some conditions are always enabled, as indicated below.

Argument	Bit	Value	Description
<i>outoff</i>	0	1	Current limit
	1	2	Voltage limit (<i>always enabled</i>)
	2	4	Photodiode current limit
	3	8	Photodiode power limit
	4	16	Interlock disabled (<i>always enabled</i>)
	5	32	Unused
	6	64	Unused
	7	128	Laser open circuit (<i>always enabled</i>)
	8	256	Laser shorted (<i>always enabled</i>)
	9	512	Out of tolerance
	10	1024	Unused
	11	2048	Unused
	12	4096	Hardware error (<i>always enabled</i>)
	13	8192	Unused
	14	16384	Unused
	15	32768	Thermal limit exceeded (<i>always enabled</i>)

The default value for this register is 37266.

See Also LASer:ENABLE:OUTOFF?

LASer:ENABLE:OUTOFF?**Synopsis** Query the Output Off Enable register

Syntax LASer:ENABLE:OUTOFF?

Details Returns the value of the Output Off register. See the LASer:ENABLE:OUTOFF command for definition of *outoff* response value.

See Also LASer:ENABLE:OUTOFF

LASer:EVENT?

Synopsis Query the value of the laser event register

Syntax LASer:EVENT?

Details Returns the laser event register.

Response	Bit	Value	Description
<i>events</i>	0	1	Current limit
	1	2	Voltage limit
	2	4	Photodiode current limit
	3	8	Photodiode power limit
	4	16	Interlock state changed
	5	32	Unused
	6	64	Unused
	7	128	Laser open circuit
	8	256	Laser short circuit
	9	512	Out of tolerance changed state
	10	1024	Output changed state
	11	2048	New data
	12	4096	Unused
	13	8192	Unused
	14	16384	Unused
15	32768	Unused	

After reading the event register, the event register is set to zero.

See Also LASer:ENABLE:EVENT

LASer:F

Synopsis Set the frequency of pulses in QCW mode

Syntax LASer:F *frequency*

Details Changes the frequency when in QCW mode. A new duty cycle value will be computed based on the frequency and pulse width (the pulse width is not changed). If the resulting duty cycle is outside the allowable range, an E-201 error will be generated and the command ignored.

Argument	Description
<i>frequency</i>	Frequency, in hertz

See Also LASer:F?

LASer:F?**Synopsis** Query the pulse frequency**Syntax** LASer:F?**Details** Returns the value of the frequency. See LASer:F for a definition of the *frequency* response value.**See Also** LASer:F

LASer:INC**Synopsis** Increment the laser set point**Syntax** LASer:INC *steps* [, *time*]**Details** The LASer:INC command uses the step size defined with the LASer:STEP command to increment the laser set point. If the *time* parameter is omitted, then the set point is immediately incremented *steps* times the step size. If the *time* parameter is included, then the instrument will still increment *steps* times the step size, but will pause at each step for *time* milliseconds.

Argument	Value	Description
<i>steps</i>	1 to 65000	Number of steps to increment
<i>time</i>	0 to 65000	Number of milliseconds to pause between each step

See Also LASer:DEC, LASer:STEP

LASer:LDI (I)**Synopsis** Set the laser current set point**Syntax** LASer:LDI *setpoint***Details** The LASer:LDI sets the laser current set point. An error will be generated if the value of *setpoint* is greater than the current limit.

Argument	Value	Description
<i>setpoint</i>	0 to I_{lim}	Set the current set point in milliamps

See Also LASer:LDI?, LASer:LIMit:LDI, LASer:SET:LDI

LASer:LDI? (I?)**Synopsis** Query the actual laser current**Syntax** LASer:LDI?**Details** Returns the actual (measured) laser current.

Response	Description
<i>current</i>	The actual (measured) current in milliamps

See Also LASer:LDI

LASer:LDV**Synopsis** Set the laser voltage set point**Syntax** LASer:LDV *setpoint***Details** The LASer:LDV sets the laser voltage set point.

Argument	Value	Description
<i>setpoint</i>	0 to V_{limit}	Set the voltage set point in volts

See Also LASer:LIMit:LDV?, LASer:LDV?, LASer:SET:LDV?

LASer:LDV?**Synopsis** Query the laser actual voltage**Syntax** LASer:LDV?**Details** Returns the actual laser voltage.

Response	Description
<i>voltage</i>	Actual laser voltage, in volts

See Also LASer:LIMit:LDV, LASer:LDV

LASer:LIMit:LDI (I)**Synopsis** Set the laser current limit**Syntax** LASer:LIMit:LDI *limit***Details** The LASer:LDI sets the laser current limit. If the laser current set point is greater than the value of *limit*, then laser current set point will be reduced to the value of *limit*.

Argument	Value	Description
<i>limit</i>	0 to I_{max}	Set the laser current limit in milliamps

See Also LASer:LDI, LASer:LIMit:LDI?

LASer:LIMit:LDI? (I?)**Synopsis** Query the laser current limit**Syntax** LASer:LIMit:LDI?**Details** Returns the value of the laser current limit. See LASer:LIMit:LDI for a definition of the *limit* response value.**See Also** LASer:LIMit:LDI

LASer:LIMit:LDV**Synopsis** Set the laser voltage limit**Syntax** LASer:LIMit:LDV *limit*

Details The LASer:LIMit:LDV sets the laser voltage limit. If the laser voltage set point is greater than the value of *limit*, then laser voltage set point will be reduced to the value of *limit*.

Argument	Value	Description
<i>limit</i>	0 to V_{\max}	Set the laser voltage limit in volts

See Also LASer:LIMit:LDV?, LASer:LDV

LASer:LIMit:LDV?

Synopsis Query the laser voltage limit

Syntax LASer:LIMit:LDV?

Details Returns the value of the laser voltage limit. See LASer:LIMit:LDV for a definition of the *limit* response value.

See Also LASer:LIMit:LDV, LASer:LDV

LASer:LIMit:MDI (IPD)

Synopsis Set the photodiode current limit

Syntax LASer:LIMit:MDI *limit*

Details The LASer:LIMit:MDI sets the photodiode current limit. If the photodiode current set point is greater than the value of *limit*, then photodiode current set point will be reduced to the value of *limit*.

Argument	Value	Description
<i>limit</i>	0 to $I_{m_{\max}}$	Set the photodiode current limit in microamps

See Also LASer:LIMit:MDI?, LASer:MDI

LASer:LIMit:MDI? (IPD?)

Synopsis Query the photodiode current limit

Syntax LASer:LIMit:MDI?

Details Returns the value of the photodiode current limit. See LASer:LIMit:MDI for a definition of the *limit* response value.

See Also LASer:LIMit:MDI, LASer:MDI

LASer:LIMit:MDP (Ppd)

Synopsis Set the photodiode power limit

Syntax LASer:LIMit:MDP *limit*

Details The LASer:LIMit:MDP sets the photodiode power limit. If the photodiode power set point is greater than the value of *limit*, then photodiode power set point will be reduced to the value of *limit*.

Argument	Value	Description
<i>limit</i>	0 to P_{max}	Set the photodiode power limit in milliwatts

See Also LASer:LIMit:MDP?, LASer:MDP

LASer:LIMit:MDP? (Ppd?)

Synopsis Query the photodiode power limit

Syntax LASer:LIMit:MDP?

Details Returns the value of the photodiode power limit. See LASer:LIMit:MDP for a definition of the *limit* response value.

See Also LASer:LIMit:MDP, LASer:MDP

LASer:MDI (IPD)

Synopsis Set the photodiode current set point

Syntax LASer:MDI *setpoint*

Details The LASer:MDI sets the photodiode current set point.

Argument	Value	Description
<i>setpoint</i>	0 to I_{limit}	Set the photodiode current set point in microamps

See Also LASer:LIMit:MDI?, LASer:MDI?, LASer:SET:MDI?

LASer:MDI? (IPD?)

Synopsis Query the actual photodiode current

Syntax LASer:MDI?

Details Returns the actual photodiode current.

Response	Description
<i>PDcurrent</i>	Actual photodiode current, in microamps

See Also LASer:LIMit:MDI, LASer:MDI

LASer:MDP (Ppd)

Synopsis Set the photodiode power set point

Syntax LASer:MDP *setpoint*

Details The LASer:MDP sets the photodiode power set point.

Argument	Value	Description
<i>setpoint</i>	0 to $P_{o_{limit}}$	Set the photodiode power set point in milliwatts

See Also LASer:LIMit:MDP?, LASer:MDP?, LASer:SET:MDP?

LASer:MDP? (Ppd?)**Synopsis** Query the actual photodiode power**Syntax** LASer:MDP?**Details** Returns the actual photodiode power.

Response	Description
<i>PDpower</i>	Actual photodiode power, in milliwatts

See Also LASer:LIMit:MDP, LASer:MDP

LASer:MODE?**Synopsis** Query operational mode**Syntax** LASer:MODE?**Details** Returns the control mode of the LaserSource.

Response	Value	Description
<i>mode</i>	ILBW	Laser current control (Io/ACC) mode, low bandwidth
	IHBW	Laser current control (Io HiBW) mode, high bandwidth
	PULSE	Laser current control (Io Pulse) mode, QCW internal trigger
	TRIG	Laser current control (Io Ext Trig) mode, QCW external trigger
	BURST	Laser current control (Io Burst) mode, QCW burst
	MDI	Photodiode current control (Im/AMC) mode
	MDP	Photodiode power control (Po/APC) mode
	LDV	Laser voltage control (Vf/AVC) mode

See Also LASer:MODE:ILBW, LASer:MODE:IHBW, LASer:MODE:PULSE, LASer:MODE:TRIG, LASer:MODE:BURST, LASer:MODE:MDI, LASer:MODE:MDP; LAS:MODE:LDV

LASer:MODE:BURST**Synopsis** Set operational mode to laser current control, QCW burst.**Syntax** LASer:MODE:BURST**Details** Switches the LaserSource to laser current control mode, QCW burst. Pulses are triggered with a LASer:TRIGGER command or via the external trigger. The number of pulses is controlled by the LASer:QCWCOUNT command. If the mode is being changed and the laser output is on, the laser output will be turned off and an error generated.**See Also** LASer:MODE?, LASer:QCWCOUNT, LASer:TRIGGER

LASer:MODE:ICW (CW)**Synopsis** Set operational mode to laser current control, continuous wave**Syntax** LASer:MODE:ICW**Details** This command is identical to LASer:MODE:ILBW, and is included for compatibility reasons. If a LASer:MODE? query is done after this command is issued, "ILBW" will be returned.

See Also LASer:MODE:ILBW

LASer:MODE:ILBW (I)

Synopsis Set operational mode to laser current control, low bandwidth

Syntax LASer:MODE:ILBW

Details Switches the LaserSource to laser current control mode, low bandwidth. If the mode is being changed and the laser output is on, the laser output will be turned off and an error generated.

See Also LASer:MODE?

LASer:MODE:IHBW

Synopsis Set operational mode to laser current control, high bandwidth

Syntax LASer:MODE:IHBW

Details Switches the LaserSource to laser current control mode, high bandwidth. If the mode is being changed and the laser output is on, the laser output will be turned off and an error generated.

See Also LASer:MODE?

LASer:MODE:MDI (IPD)

Synopsis Set operational mode to photodiode current control

Syntax LASer:MODE:MDI

Details Switches the LaserSource to photodiode current control mode. If the mode is being changed and the laser output is on, the laser output will be turned off and an error generated.

See Also LASer:MODE?

LASer:MODE:MDP (Ppd)

Synopsis Set operational mode to photodiode power control

Syntax LASer:MODE:MDP

Details Switches the LaserSource to photodiode power control mode. If the mode is being changed and the laser output is on, the laser output will be turned off and an error generated.

See Also LASer:MODE?

LASer:MODE:LDV

Synopsis Set operational mode to laser voltage control

Syntax LASer:MODE:LDV

Details Switches the LaserSource to laser voltage control mode. If the mode is being changed and the laser output is on, the laser output will be turned off and an error generated.

See Also LASer:MODE?

LASer:MODE:PULSE

Synopsis Set operational mode to laser current control, QCW mode, internal trigger

Syntax LASer:MODE:PULSE

Details Switches the LaserSource to laser current control mode, QCW. Pulses are internally generated using the pulse width, duty cycle, and frequency settings. If the mode is being changed and the laser output is on, the laser output will be turned off and an error generated.

See Also LASer:MODE?

LASer:MODE:TRIG

Synopsis Set operational mode to laser current control, QCW mode, external trigger

Syntax LASer:MODE:TRIG

Details Switches the LaserSource to laser current control mode, QCW. Pulses are externally triggered with the TRIGIN signal. Each rising edge of TRIG in generates one pulse. Pulse width is determined by the pulse width setting. Duty cycle and frequency are ignored. If the mode is being changed and the laser output is on, the laser output will be turned off and an error generated.

See Also LASer:MODE?

LASer:OUTput

Synopsis Set the laser output state

Syntax LASer:OUTput *state*

Details Turns the laser output on or off.

Argument	Value	Description
<i>state</i>	0	Turn the output off
	1	Turn the output on

See Also LASer:OUTput?

LASer:OUTput?

Synopsis Query the laser output state

Syntax LASer:OUTput?

Details Returns the value of the laser current limit. See LASer:OUT for a definition of the *state* response value.

See Also LASer:OUTput

LASer:PDBias**Synopsis** Set the photodiode bias voltage set point**Syntax** LASer:PDBias *voltage***Details** Sets the photodiode bias voltage set point.

Argument	Value	Description
<i>voltage</i>	0 to 5	Negative voltage bias set point, in volts

See Also LASer:PDBias?

LASer:PDBias?**Synopsis** Query the photodiode bias voltage set point**Syntax** LASer:PDBias?**Details** Returns the value of the photodiode bias voltage set point. See LASer:PDBias for a definition of the *voltage* response value.**See Also** LASer:PDBias

LASer:PW?**Synopsis** Query the pulse width**Syntax** LASer:PW?**Details** Returns the pulse width, in milliseconds. See LASer:PWF and LASer:PWP for more information on setting the pulse width.**See Also** LASer:PWF, LASer:PWP

LASer:PWF**Synopsis** Set the pulse width while holding the frequency constant**Syntax** LASer:PWF *pulsewidth***Details** Changes the pulse width when in QCW mode. A new duty cycle value will be computed based on the frequency and pulse width. If the resulting duty cycle is outside the allowable range, an E-201 error will be generated and the command ignored.

Argument	Description
<i>pulsewidth</i>	Pulse width, in milliseconds

See Also LASer:PW?, LASer:PWP, LASer:QCWHOLD

LASer:PWP**Synopsis** Set the pulse width while holding the duty cycle constant**Syntax** LASer:PWP *pulsewidth*

Details Changes the pulse width when in QCW mode. A new frequency value will be computed based on the duty cycle and pulse width. If the resulting frequency is outside the allowable range, an E-201 error will be generated and the command ignored.

Argument	Description
<i>pulsewidth</i>	Pulse width, in milliseconds

See Also LASer:PW?, LASer:PWF, LASer:QCWHOLD

LASer:QCWHOLD

Synopsis Set the QCW (pulse) hold setting for pulse width adjustments in the user interface

Syntax LASer:QCWHOLD *hold*

Details Sets which value to hold constant (frequency or duty cycle) when making pulse width adjustments in the user interface.

This command has no effect on the LASer:PWF and LASer:PWP commands, as they explicitly define which will be held constant. It only affects the pulse width adjustments done through the front panel.

Argument	Value	Description
<i>hold</i>	0	Hold frequency constant
	1	Hold duty cycle constant

See Also LASer:QCWHOLD?

LASer:QCWHOLD?

Synopsis Queries the QCW (pulse) hold setting for pulse width adjustments in the user interface

Syntax LASer:QCWHOLD?

Details Returns the value of the hold constant. See LASer:QCWHOLD for a definition of the *hold* response value.

See Also LASer:QCWHOLD

LASer:QCWCOUNT

Synopsis Set the QCW burst count

Syntax LASer:QCWCOUNT *count*

Details Sets how many pulses will be generated for each burst when operating in IO (Burst) mode.

Argument	Value	Description
<i>count</i>	1 – 60,000	Number of pulses per burst

See Also LASer:QCWCOUNT?

LASer: QCWCOUNT?

Synopsis Queries the number of pulses generated for each burst when operating in Io (Burst) mode.

Syntax LASer:QCWCOUNT?

Details Returns the pulse count.. See LASer:QCWCOUNT for a definition of the *count* response value.

See Also LASer:QCWCOUNT

LASer:SET:LDI? (I?)

Synopsis Query the laser current set point

Syntax LASer:SET:LDI?

Details Returns the value of the laser current voltage set point. See LASer:LDI for a definition of the *setpoint* response value.

See Also LASer:LDI

LASer:SET:MDI? (IPD?)

Synopsis Query the photodiode current set point

Syntax LASer:SET:MDI? (IPD?)

Details Returns the value of the photodiode current set point. See LASer:MDI for a definition of the *setpoint* response value.

See Also LASer:MDI

LASer:SET:MDP? (Ppd?)

Synopsis Query the photodiode power set point

Syntax LASer:SET:MDP? (Ppd?)

Details Returns the value of the photodiode power set point. See LASer:MDP for a definition of the *setpoint* response value.

See Also LASer:MDP

LASer:SET:LDV?

Synopsis Query the laser voltage set point

Syntax LASer:SET:LDV?

Details Returns the value of the laser voltage set point. See LASer:LDV for a definition of the *setpoint* response value.

See Also LASer:LDV

LASer:STB?**Synopsis** Query the laser status byte**Syntax** LASer:STB?**Details** Returns a summary of the enabled conditions within the laser condition and event registers. These bits mirror the bits in the Status Byte Register.

Argument	Bit	Value	Description
<i>status</i>	2	4	Event status register summary
	3	8	Condition status register summary

The values are additive, so a return value of 0, 4, 8, or 12 is possible.

See Also *STB?, LASer:COND?, LASer:ENAB:COND, LASer:ENABLE:EVENT, LASer:EVEnt?

LASer:STEP**Synopsis** Set laser step size**Syntax** LASer:STEP *size***Details** The command sets the laser step size used by the LASer:INC or LASer:DEC commands.

Argument	Value	Description
<i>size</i>	1 to 65000	Step size

A step of 1 equates to 0.01mA, 0.01mW, or 1 μ A, depending on the control mode.

See Also LASer:DEC, LASer:INC, LASer:STEP?

LASer:STEP?**Synopsis** Query the laser step size**Syntax** LASer:STEP?**Details** Returns the value of the laser step size. See LASer:STEP for a definition of the *size* response value.**See Also** LASer:STEP

LASer:TOLerance**Synopsis** Set the laser tolerance criteria**Syntax** LASer:TOLerance *tolerance, time***Details** The LASer:TOLerance command allows control over when the output of the laser driver is considered in tolerance (or stable), in order to satisfy the tolerance condition of the operation complete definition. When used in conjunction with the *WAI command, it can control when the next command is processed, delaying processing until the output stabilizes at its set point.

Argument	Value	Description
<i>tolerance</i>	0.01 to I_{max}	Current tolerance, in milliamps
<i>time</i>	0.1 to 50	Time window in seconds

To be considered in tolerance, the measured current must be within the set point plus or minus the *tolerance* value (the tolerance window) for *time* seconds. Any time it leaves the tolerance window, the timer will reset to zero and begin counting the next time it enters the tolerance window.

In photodiode current mode, the *tolerance* is fixed at 50 μ A. In photodiode power mode, *tolerance* is fixed at 50mW. In voltage control mode, *tolerance* is fixed at 50mV.

See Also LASer:TOLerance?

LASer:TOLerance?

Synopsis Query the laser tolerance criteria

Syntax LASer:TOLerance?

Details Returns the value of the laser tolerance criteria. See LASer:TOLerance for a definition of the *tolerance* and *time* response values.

See Also LASer:TOLerance

LASer:TRIGger

Synopsis Generates a single trigger

Syntax LASer:TRIGger

Details The LASer:TRIGger command is used to generate a trigger into the QCW system, and acts exactly like a single rising edge on the Trigger Input BNC. See the manual section on QCW operation for more information on triggering.

LASer:VSENSE

Synopsis Select internal or external voltage sense

Syntax LASer:VSENSE *select*

Details For instruments that support selectable internal/external voltage sense, this command selects internal or external sense.

Argument	Value	Description
<i>select</i>	0	Internal voltage sense
	1	External voltage sense

See Also LASer:VSENSE?

LASer: VSENSE?

Synopsis Queries the voltage sense selection.

Syntax LASer:VSENSE?

Details For instruments that support selectable internal/external voltage sense, this command returns the *select* value. See the LASer:VSENSE command for more information.

See Also LASer:VSENSE

LOCAL

Synopsis Returns instrument to local mode

Syntax LOCAL

Details After issuing the LOCAL command, the remote light will go out, and front panel controls will be enabled.

MESsage

Synopsis Set the message buffer

Syntax MESsage *string*

Details Sets the internal message buffer to the value of *string*, up to a maximum of 16 characters.

See Also MESsage?

MESsage?

Synopsis Query the message buffer

Syntax MESsage?

Details Returns the value of the message buffer.

See Also MESsage

ONDELAY

Synopsis Set the laser output on delay

Syntax ONDELAY *time*

Details The ONDELAY command controls how long the laser driver will delay between the time the user or command turns on the laser out and when the output is actually turned on.

Argument	Value	Description
<i>time</i>	0 to 30000	On delay, in milliseconds

Setting the ONDELAY value to 0 disables the delay feature.

See Also ONDELAY?

ONDELAY?

Synopsis Query the laser output on delay

Syntax ONDELAY?

Details Returns the value of the laser output on delay. See the ONDELAY command for a complete definition of possible return values.

See Also ONDELAY

RADix

Synopsis Set the radix (number base)

Syntax RADix *base*

Details By default, the instrument is set to decimal number base. Changing to an alternate number base will cause queries for integer values to be returned in the specified number base.

Argument	Value	Description
<i>base</i>	BIN	Binary format (base 2)
	OCT	Octal format (base 8)
	DEC	Decimal format (base 10)
	HEX	Hexadecimal format (base 16)

See Also RADix?

RADix?

Synopsis Query the radix (number base)

Syntax RADix?

Details Returns the current number base. See the RADix command for a complete definition of possible return values.

See Also RADix

REMERR

Synopsis Set display of errors while in remote mode

Syntax REMERR *enable*

Details This command controls if the instrument will display errors while in remote mode. If set to zero, then errors will not be displayed. If set to one, errors will be displayed. Errors will always accumulate in the error queue.

Argument	Value	Description
<i>enable</i>	0	No not display errors in remote mode
	1	Display errors in remote mode

See Also REMERR?, ERR?, ERRSTR?

REMERR?

Synopsis Query the display of errors while in remote mode

Syntax REMERR?

Details Returns the current REMERR setting. See the REMERR command for a complete definition of possible return values.

See Also REMERR

TEC:AUTOTUNE

Synopsis Start the AutoTune process

Syntax TEC:AUTOTUNE *temperature*

Details The TEC:AUTOTUNE command is used to start the AutoTune process, using the *temperature* parameter as the AutoTune point. The current and temperature limits should be properly setup prior to starting AutoTune.

Argument	Description
<i>temperature</i>	AutoTune test point, in °C

See the AutoTune section in the user's manual for further information.

See Also TEC:AUTOTUNE?

TEC:AUTOTUNE?

Synopsis Query the AutoTune result

Syntax TEC:AUTOTUNE?

Details Returns the result of the last AutoTune process.

Response	Description
0	No AutoTune has been performed since last power-up
1	AutoTune is process
2	Last AutoTune failed
3	Last AutoTune successful

The TEC:AUTOTUNESTATE? provides additional details during the AutoTune process.

See Also TEC:AUTOTUNE, TEC:AUTOTUNESTATE?

TEC:AUTOTUNESTATE?

Synopsis Query the AutoTune result

Syntax TEC:AUTOTUNESTATE?

Details Returns the progress of the AutoTune process.

Response	Description
0	No AutoTune in process
100	AutoTune initializing
200	AutoTune in P ramp phase
300	AutoTune in P stabilize phase
400	AutoTune in I ramp phase
500	AutoTune Complete

See Also TEC:AUTOTUNE

TEC:CABLER

Synopsis Set the cable resistance

Syntax TEC:CABLER *resistance*

Details The TEC:CABLER command can be used to set the cable resistance, which is then used to remove voltage drops from the TEC voltage measurement.

Argument	Description
<i>resistance</i>	Resistance of the cable, in ohms.

See the user's manual for additional information on using this setting.

See Also TEC:CABLER?, TEC:VTE?

TEC:CABLER?

Synopsis Query the cable resistance value

Syntax TEC:CABLER?

Details Returns the value of the cable resistance value. See TEC:CABLER for a definition of the *resistance* response value.

See Also TEC:CABLER

TEC:CABLETYPE?

Synopsis Returns the cable type (high or low current)

Syntax TEC:CABLETYPE?

Details Returns the current rating for the cable plugged into the instrument.

Response	Description
0	8A cable
1	10A cable

TEC:COND?

Synopsis Query TEC condition

Syntax TEC:COND?

Details Returns the TEC condition register.

Response	Bit	Value	Description
<i>conditions</i>	0	1	Current limit
	1	2	Voltage limit
	2	4	Sensor limit
	3	8	Temperature high limit
	4	16	Temperature low limit
	5	32	Sensor shorted
	6	64	Sensor open
	7	128	TEC open circuit
	8	256	Unused
	9	512	Out of tolerance
	10	1024	Output on
	11	2048	Unused
	12	4096	Unused
	13	8192	Unused
	14	16384	Unused
15	32768	Unused	

See Also TEC:ENABLE:COND, *STB?

TEC:CONST

Synopsis Set sensor temperature conversion constants

Syntax TEC:CONST A, B [, C [, R0]]

Details The TEC:CONST command sets the sensor constants for conversion of the sensor value to temperature. The number of parameters are dependent on the active sensor.

For the thermistor sensors, A, B, and C are used as constants for the Steinhart-Hart equation for conversion of resistance to temperature. By default, the thermistor constants are set to those for a BetaTHERM 10K3A1 thermistor.

For LM335 and AD590 sensors, A is used as a slope correction (M) term and B is an offset (B) term. By default, A is set to one and B is set to zero.

For RTD sensors, A, B, C, and R0 are used as constants for the RTD to temperature equation. By default, the RTD constants are set to the standard 100Ω Laboratory values.

Argument	Value	Description
<i>For Thermistors:</i>		
A	± 9.9999	First Steinhart-Hart constant (x 10 ⁻³)
B	± 9.9999	Second Steinhart-Hart constant (x 10 ⁻⁴)
C	± 9.9999	Third Steinhart-Hart constant (x 10 ⁻⁷)
<i>For LM335:</i>		
A	± 9.9999	Slope term
B	± 99.9999	Offset term, in °C
<i>For AD590:</i>		
A	± 9.9999	Slope term
B	± 99.9999	Offset term, in °C
<i>For RTD:</i>		
A	± 9.9999	First RTD constant (x 10 ⁻³)
B	± 9.9999	Second RTD constant (x 10 ⁻⁶)
C	± 9.9999	Third RTD constant (x 10 ⁻¹²)
R0	10 to 220	Nominal resistance at 0°C (Ω)

Some vendors may refer to A, B, and C as C1, C2, & C3.

For additional information, see the sensor sections in the user's manual.

See Also TEC:CONST?

TEC:CONST?

Synopsis Query sensor temperature conversion constants

Syntax TEC:CONST?

Details Returns the sensor temperature conversion constants. See the TEC:CONST command for a complete definition of the A, B, C, and R0 response values.

See Also TEC:CONST

TEC:DEC

Synopsis Decrement the TEC set point

Syntax TEC:DEC *steps*

Details The TEC:DEC command uses the step size defined with the TEC:STEP command to decrement the TEC set point. The set point is decremented *steps* times the step size.

Argument	Value	Description
<i>steps</i>	1 to 65000	Number of steps to decrement

See Also TEC:INC, TEC:STEP

TEC:DISPlay

Synopsis Set the display enable state

Syntax TEC:DISPlay *enable*

Details The TEC:DISplay command can be used to completely lock out local operation of the instrument and display “Display Disabled” instead of the normal display.

Argument	Value	Description
<i>enable</i>	0	Disables the display and front panel
	1	Enables the display and front panel

Once the display is disabled, the front panel is completely locked out. The only way to restore functionality to the front panel is to issue a “TEC:DISplay 1” or cycle power on the unit.

See Also TEC:DISplay?

TEC:DISplay?

Synopsis Query the display enable state

Syntax TEC:DISplay?

Details Returns the value of the laser display enable state. See TEC:DISplay for a definition of the *enable* response value.

See Also TEC:DISplay

TEC:ENABle:COND

Synopsis Set TEC Condition Enable register

Syntax TEC:ENABle:COND *conditions*

Details Enables reporting of selected conditions to the Status Byte Register. See the TEC:COND command for a definition of the *conditions* parameter. The default value for this register is 0.

See Also TEC:ENABle:COND?, TEC:COND?

TEC:ENABle:COND?

Synopsis Query TEC Condition Enable register

Syntax TEC:ENABle:COND?

Details Returns the value of the TEC Condition Enable register. See the TEC:COND command for a definition of the *conditions* response.

See Also TEC:ENABle:COND, TEC:COND?

TEC:ENABle:EVEnt

Synopsis Set TEC Event Enable register

Syntax TEC:ENABle:EVEnt *events*

Details Enables reporting of selected events to the Status Byte Register. See the TEC:EVEnt command for a definition of the *events* parameter. The default value for this register is 0.

See Also see also TEC:ENABLE:EVEnt?; TEC:EVE?

TEC:ENABLE:EVEnt?

Synopsis Query TEC Event Enable register

Syntax TEC:ENABLE:EVEnt?

Details Returns the value of the TEC Event Enable register. See the TEC:EVEnt command for a definition of the *events* response.

See Also TEC:ENABLE:EVEnt, TEC:EVEnt?

TEC:ENABLE:OUTOFF

Synopsis Set the Output Off Enable register

Syntax TEC:ENABLE:OUTOFF *outoff*

Details The Output Off register controls what conditions will cause the TEC output to be turned off. Some conditions are always enabled, as indicated below.

Argument	Bit	Value	Description
<i>outoff</i>	0	1	Current limit
	1	2	Voltage limit
	2	4	Sensor limit
	3	8	Temperature high limit
	4	16	Temperature low limit
	5	32	Unused
	6	64	Sensor open
	7	128	TEC open circuit
	8	256	Sensor type change (<i>always enabled</i>)
	9	512	Out of tolerance
	10	1024	Sensor short
	11	2048	Unused
	12	4096	Unused
	13	8192	Unused
	14	16384	Unused
	15	32768	Unused

The default value for this register is 1496, indicated by the bold elements above

See Also TEC:ENABLE:OUTOFF?

TEC:ENABLE:OUTOFF?

Synopsis Query the Output Off Enable register

Syntax TEC:ENABLE:OUTOFF?

Details Returns the value of the Output Off register. See the TEC:ENABLE:OUTOFF command for definition of *outoff* response value.

See Also TEC:ENABLE:OUTOFF

TEC:EVEnt?**Synopsis** Query the TEC event register**Syntax** TEC:EVEnt?**Details** Returns the TEC event register.

Response	Bit	Value	Description
<i>events</i>	0	1	Current limit
	1	2	Voltage limit
	2	4	Sensor limit
	3	8	Temperature high limit
	4	16	Temperature low limit
	5	32	Sensor shorted
	6	64	Sensor open
	7	128	TEC open circuit
	8	256	Sensor type change
	9	512	Out of tolerance changed state
	10	1024	Output changed state
	11	2048	Unused
	12	4096	Unused
	13	8192	Unused
	14	16384	Unused
	15	32768	Unused

After reading the event register, the event register is set to zero.

See Also TEC:ENABLE:EVEnt

TEC:FAN**Synopsis** Set the external fan speed**Syntax** TEC:FAN *speed*[, *mode*[, *delay*]]**Details** Set the external fan speed.

Argument	Value	Description
<i>speed</i>	OFF, SLOW, MEDIUM, FAST, or 4.0 to 12.0	Fan speed
<i>mode</i>	1, 2, or 3	Fan delay mode: 1 = Auto (fan off when TEC off) 2 = On (fan always on) 3 = Delayed off (fan turns off <i>delay</i> minutes after TEC turned off).
<i>delay</i>	1 to 240	Minutes to delay turning fan off (only applies to 'Delayed off' <i>mode</i>).

The values SLOW, MEDIUM, and FAST correspond to 9V, 10.5V, and 12V, respectively. The fan speed can also be controlled by selecting a specific voltage to drive the fan. When use the external fan control, ensure your fan and voltage setting are properly match, or damage to the fan may occur.

Setting the *mode* to 1 will turn the fan off whenever the output is off. Setting the *mode* to 2 will turn the fan on always, regardless of the TEC output state. Setting the *mode* to 3 will delay the turning off of the fan to *delay* minutes after the TEC output is turned off.

The *delay* setting only applies when *mode* is set to 3 (delayed off), and sets the number of minutes to delay the fan from turning off after the output has been turned off.

The 5230 does not support the FAN setting.

See Also TEC:FAN?

TEC:FAN?

Synopsis Query the external fan speed setting

Syntax TEC:FAN?

Details Returns the fan speed setting. See TEC:FAN for a definition of the *speed*, *mode*, and *delay* response values.

See Also TEC:FAN

TEC:GAIN

Synopsis Set the control loop gain

Syntax TEC:GAIN *gain*

Details Set the temperature control loop gain.

Argument	Value	Description
<i>gain</i>	1, 3, 5, 10, 30, 50, 100, 300, PID	TEC gain

The 5230 does not support the PID setting.

See Also TEC:GAIN?; TEC:PID

TEC:GAIN?

Synopsis Query the control loop gain

Syntax TEC:GAIN?

Details Returns the value of control loop gain. See TEC:GAIN for a definition of the *gain* response value.

See Also TEC:GAIN; TEC:PID

TEC:HEATCOOL

Synopsis Set the heat/cool mode

Syntax TEC:HEATCOOL *mode*

Details Sets the heat/cool mode.

Argument	Value	Description
<i>mode</i>	BOTH, HEAT, COOL	Heating/cooling mode

The 5230 does not support the HEATCOOL setting.

See Also TEC:HEATCOOL?

TEC:HEATCOOL?

Synopsis Query the heat/cool mode

Syntax TEC:HEATCOOL?

Details Returns the heat/cool mode setting. See TEC:HEATCOOL for a definition of the *mode* response value.

See Also TEC:HEATCOOL

TEC:INC

Synopsis Increment the TEC set point

Syntax TEC:INC *steps*

Details The TEC: INC command uses the step size defined with the TEC:STEP command to increment the TEC set point. The set point is incremented *steps* times the step size.

Argument	Value	Description
<i>steps</i>	1 to 65000	Number of steps to increment

See Also TEC:DEC, TEC:STEP

TEC:ITE

Synopsis Set the current set point

Syntax TEC:ITE *setpoint*

Details Sets the current set point. It must be within the current limit.

Argument	Description
<i>setpoint</i>	Current set point, in amps

Not available in 5230 TECSOURCE.

See Also TEC:SET:ITE?, TEC:ITE?

TEC:ITE?

Synopsis Query the TEC current

Syntax TEC:ITE?

Details The TEC:ITE? Query returns the measured TEC current.

Response	Value	Description
<i>ite</i>	$\pm ITE_{lim}$	Measure TEC current, in amps

See Also TEC:LIMit:ITE

TEC:LIMit:ITE

Synopsis Set the TEC current limit

Syntax TEC:LIMit:ITE *limit*

Details Sets the TEC current limit.

Argument	Value	Description
<i>limit</i>	$\pm ITE_{max}$	TEC current limit, in amps

See Also TEC:ITE?, TEC:LIMit:ITE?

TEC:LIMit:ITE?

Synopsis Query the TEC current limit

Syntax TEC:LIMit:ITE?

Details Returns the value of the TEC current limit. See TEC:LIMit:ITE for a definition of the *limit* response value.

See Also TEC:LIMit:ITE

TEC:LIMit:RHI

Synopsis Set the high sensor limit

Syntax TEC:LIMit:RHI *limit*

Details Sets the high sensor limit.

Argument	Description
<i>limit</i>	High sensor limit: Thermistors in k Ω AD590s in μA LM335s in mV RTDs in Ω

See Also TEC:LIMit:RHI?

TEC:LIMit:RHI?

Synopsis Query the high sensor limit

Syntax TEC:LIMit:RHI?

Details Returns the value of the high sensor limit. See TEC:LIMit:RHI for a definition of the *limit* response value.

See Also TEC:LIMit:RHI

TEC:LIMit:RLO**Synopsis** Set the low sensor limit**Syntax** TEC:LIMit:RLO *limit***Details** Sets the low sensor limit.

Argument	Description
<i>limit</i>	Low sensor limit: Thermistors in k Ω AD590s in μ A LM335s in mV RTDs in Ω

See Also TEC:LIMit:RLO?

TEC:LIMit:RLO?**Synopsis** Query the low sensor limit**Syntax** TEC:LIMit:RLO?**Details** Returns the value of the low sensor limit. See TEC:LIMit:RLO for a definition of the *limit* response value.**See Also** TEC:LIMit:RLO

TEC:LIMit:THI**Synopsis** Set the high temperature limit**Syntax** TEC:LIMit:THI *limit***Details** Sets the high temperature limit.

Argument	Value	Description
<i>limit</i>	-99 to +250	High temperature limit, in degrees

See Also TEC:LIMit:THI?

TEC:LIMit:THI?**Synopsis** Query the high temperature limit**Syntax** TEC:LIMit:THI?**Details** Returns the value of the high temperature limit. See TEC:LIMit:THI for a definition of the *limit* response value.**See Also** TEC:LIMit:THI

TEC:LIMit:TLO**Synopsis** Set the low temperature limit

Syntax TEC:LIMit:TLO *limit*

Details Sets the low temperature limit.

Argument	Value	Description
<i>limit</i>	-99 to +250	Low temperature limit, in degrees

See Also TEC:LIMit:TLO?

TEC:LIMit:TLO?

Synopsis Query the low temperature limit

Syntax TEC:LIMit:TLO?

Details Returns the value of the low temperature limit. See TEC:LIMit:TLO for a definition of the *limit* response value.

See Also TEC:LIMit:TLO

TEC:MODE?

Synopsis Query the TEC control mode

Syntax TEC:MODE?

Details Returns the TEC control mode.

Response	Value	Description
<i>mode</i>	T	Temperature control mode
	R	Sensor control mode
	ITE	Current control mode

See Also TEC:MODE:R, TEC:MODE:T; TEC:MODE:ITE

TEC:MODE:ITE

Synopsis Set current control mode

Syntax TEC:MODE:ITE

Details Changes the set point to amps and sensor measurement to degrees Celsius.

Not available in 5230 TECSource.

See Also TEC:MODE?, TEC:MODE:T;TEC:MODE:R

TEC:MODE:R

Synopsis Set sensor control mode

Syntax TEC:MODE:R

Details Changes the set point and sensor measurement to ohms.

See Also TEC:MODE?, TEC:MODE:T, TEC:MODE:ITE

TEC:MODE:T**Synopsis** Set temperature control mode**Syntax** TEC:MODE:T**Details** Changes the set point and sensor measurement to degrees Celsius.**See Also** TEC:MODE?, TEC:MODE:R, TEC:MODE:ITE

TEC:MOUNT**Synopsis** Set the mount type**Syntax** TEC:MOUNT *mount***Details** Set the mount type to preset controller to mount's operating parameters.

Argument	Value	Description
<i>mount</i>	204, 214, 224, 242, USER	Mount type

Not available in 5230 TECSource.

See Also TEC:MOUNT?

TEC:MOUNT?**Synopsis** Query the mount type**Syntax** TEC:MOUNT?**Details** Returns the mount type. See TEC:MOUNT for a definition of the *mount* response value.**See Also** TEC:MOUNT

TEC:OUTput**Synopsis** Set the TEC output state**Syntax** TEC:OUTput *state***Details** Turns the TEC output on or off.

Argument	Value	Description
<i>state</i>	0	Turn the output off
	1	Turn the output on

See Also TEC:OUTput?

TEC:OUTput?**Synopsis** Query the TEC output state**Syntax** TEC:OUTput?**Details** Returns the TEC output state. See TEC:OUT for a definition of the *state* response value.

See Also TEC:OUTput

TEC:PID

Synopsis Sets the PID parameters

Syntax TEC:PID *p* [*i* [*d*]]

Details Sets PID parameters of the control loop when the GAIN is set to PID.

Argument	Description
<i>p</i>	The Proportional term, can be from 0 to 10
<i>i</i>	The Integral term, can be from 0 to 10
<i>d</i>	The Derivative term, can be from 0 to 10

The *i* and *d* values can be omitted, such that value commands might be:

TEC:PID 1

TEC:PID 1,2

TEC:PID 1,2,3

But the following is not valid:

TEC:PID 1,,3

See Also TEC:PID?, TEC:GAIN

TEC:PID?

Synopsis Queries the TEC PID parameters

Syntax TEC:PID?

Details Returns the TEC PID parameters used when GAIN is set to PID.

See Also TEC:PID, TEC:GAIN

TEC:R

Synopsis Set the sensor set point

Syntax TEC:R *setpoint*

Details Sets the sensor set point. It must be within the low and high sensor limits.

Argument	Description
<i>setpoint</i>	Sensor set point: Thermistors in k Ω AD590s in μ A LM335s in mV RTDs in Ω

See Also TEC:SET:R?

TEC:R?

Synopsis Query the actual sensor value

Syntax TEC:R?

Details Returns the actual (measured) sensor value.

Response	Description
<i>sensor value</i>	Actual sensor value: Thermistors in k Ω AD590s in μ A LM335s in mV RTDs in Ω

See Also TEC:R

TEC:SENsor

Synopsis Set the sensor type

Syntax TEC:SENsor *sensor*

Details Sets the sensor type.

Argument	Value	Description
<i>sensor</i>	1	100uA Thermistor
	2	10uA Thermistor
	3	LM335
	4	AD590
	5	RTD
	6	RTD (4-wire sense)
	7	1mA Thermistor

The 5230 only supports the 10uA Thermistor setting (TEC:SENS 2). Any other value will generate an error.

See Also TEC:SENsor?

TEC:SENsor?

Synopsis Query the sensor type

Syntax TEC:SENsor?

Details Returns the active sensor. See the TEC:SENsor command for a definition of the *sensor* response.

See Also TEC:SENsor

TEC:SET:ITE?

Synopsis Query current set point

Syntax TEC:SET:ITE?

Details Returns the current set point. See TEC:ITE for a definition of the *setpoint* response value.

Not available in 5230 TECSorce.

See Also TEC:ITE

TEC:SET:R?

Synopsis Query sensor set point

Syntax TEC:SET:R?

Details Returns the sensor set point. See TEC:R for a definition of the *setpoint* response value.

See Also TEC:R

TEC:SET:T?

Synopsis Query temperature set point

Syntax TEC:SET:T?

Details Returns the temperature set point. See TEC:T for a definition of the *setpoint* response value.

See Also TEC:T

TEC:STB?

Synopsis Query the TEC status byte

Syntax TEC:STB?

Details Returns a summary of the enabled conditions within the TEC condition and event registers. These bits mirror the bits in the Status Byte Register.

Response	Bit	Value	Description
<i>status</i>	0	1	Event status register summary
	1	2	Condition status register summary

The values are additive, so a return value of 0, 1, 2, or 3 is possible.

See Also *STB?, TEC:COND?, TEC:ENAB:COND, TEC:ENABLE:EVENT, TEC:EVENT?

TEC:STEP

Synopsis Set TEC step size

Syntax TEC:STEP *size*

Details The command sets the TEC step size used by the TEC:DEC or TEC:INC commands.

Argument	Value	Description
<i>size</i>	1 to 65000	Step size

The *size* value corresponds to the resolution of the set point, regardless of control mode or sensor. For example, if the temperature resolution is 0.01°C, then a step of 1 would mean a change of 0.01°C. Likewise, for example, RTDs, which typically have a display resolution of 0.01Ω, a step of one would mean a change of 0.01Ω.

See Also TEC:DEC, TEC:INC, TEC:STEP?

TEC:STEP?

Synopsis Query TEC step size

Syntax TEC:STEP?

Details Returns the TEC step size. See TEC:STEP for a definition of the *size* response value.

See Also TEC:STEP

TEC:T

Synopsis Set the temperature set point

Syntax TEC:T *setpoint*

Details Sets the temperature set point. It must be within the low and high temperature limits.

Argument	Description
<i>setpoint</i>	Temperature set point, in degrees Celsius

See Also TEC:SET:T?

TEC:T?

Synopsis Query the actual temperature

Syntax TEC:T?

Details Returns the actual (measured) sensor temperature.

Response	Description
<i>temperature</i>	Actual sensor temperature, in degrees Celsius

See Also TEC:T

TEC:TOLerance

Synopsis Set the TEC tolerance criteria

Syntax TEC:TOLerance *tolerance, time*

Details The TEC:TOLerance command allows control over when the output of the TECSource is considered in tolerance (or stable), in order to satisfy the tolerance condition of the operation complete definition. When used in conjunction with the *WAI command, it can control when the next command is processed, delaying processing until the output stabilizes at its set point.

Argument	Value	Description
<i>tolerance</i>	0.01 to 10	Tolerance, in °C
<i>time</i>	0.1 to 50	Time window in seconds

To be considered in tolerance, the measured temperature must be within the set point plus or minus the *tolerance* value (the tolerance window) for *time* seconds. Any time it

leaves the tolerance window, the timer will reset to zero and begin counting the next time it enters the tolerance window.

In sensor control mode, the *tolerance* is fixed at 0.01k Ω (100 μ A thermistor), 0.1k Ω (10 μ A thermistor), 0.1 μ A (AD590), 1mV (LM335), and 0.1 Ω (RTD).

See Also TEC:TOLerance?

TEC:TOLerance?

Synopsis Query the TEC tolerance criteria

Syntax TEC:TOLerance?

Details Returns the value of the TEC tolerance criteria. See TEC:TOLerance for a definition of the *tolerance* and *time* response values.

See Also TEC:TOLerance

TEC:V?

Synopsis Query the actual TEC voltage

Syntax TEC:V?

Details Returns the actual (measured) TEC voltage.

Response	Description
<i>voltage</i>	Actual TEC voltage, in volts

See Also TEC:T

TERM

Synopsis Set response terminator

Syntax TERM *terminator*

Details This command controls the termination characters used for responses to queries.

Argument	Value	Description
<i>terminator</i>	0 or 1	<CR><LF>
	2 or 3	<CR>
	4 or 5	<LF>
	6 or 7	no terminator

See Also TERM?

TERM?

Synopsis Query response terminator

Syntax TERM?

Details Returns the current response terminator setting. See the TERM command for a complete definition of possible return values.

See Also TERM

TERMINAL

Synopsis Set terminal mode

Syntax TERMINAL *enable*

Details The command controls the echo of characters back to the PC, typically used when manually controlling the instrument via a terminal software package such as Hyperterminal.

Argument	Value	Description
<i>enable</i>	0	Echo disabled
	1	Echo enabled

See Also TERMINAL?

TERMINAL?

Synopsis Query terminal mode

Syntax TERMINAL?

Details Returns the current response terminal mode setting. See the TERMINAL command for a complete definition of possible return values.

See Also TERMINAL

TIME?

Synopsis Query run time

Syntax TIME?

Details Returns the elapsed time since the unit has been turned on. Format is in HH:MM:SS.ss, where HH is hours, MM is minutes, SS is seconds, and ss is hundredths of a second.

See Also TIMER?

TIMER?

Synopsis Query time since last TIMER?

Syntax TIMER?

Details Returns the elapsed time since the last TIMER? query was received, or, if this is the first TIMER? query, the time since unit has been turned on. Format is in HH:MM:SS.ss, where HH is hours, MM is minutes, SS is seconds, and ss is hundredths of a second.

See Also TIME?

Error Messages

Error Code	Description	Cause
E-003	Factory EEPROM Error	The factory section of the EEPROM, which contains the calibration data and system configuration systems, is corrupted. The unit must be returned to the factory for repair.
E-004	User EEPROM Error	The user settings section of the EEPROM, which contains user editable set points, modes, etc., is corrupted. To repair the corruption, the unit was reset to factory defaults.
E-005	User Reset EEPROM	The user reset the unit to factory defaults. This is a notification message only.
E-006	User EEPROM Failed	The user settings section of the EEPROM, which contains user editable set points, modes, etc., is corrupted. The corruption cannot be repaired automatically, please contact the factory for assistance.
E-007	Preset EEPROM Failed	The preset settings section of the EEPROM, which contains user section defaults, is corrupted. The corruption cannot be repaired automatically, please contact the factory for assistance.
E-008	User EEPROM Failed	The user settings section of the EEPROM, which contains user editable set points, modes, etc., is corrupted, and the preset section cannot restore it. The corruption cannot be repaired automatically, please contact the factory for assistance.
E-009	Output Disabled	Added to every ERR? or ERRSTR? query when the EEPROM is corrupted. The output cannot be turned on until the corruption has been repaired.
E-100	General Error	The error code is non-specific, and is generally used when no other error code is suitable.
E-102	Message too long	The message is too long to process (USB/Serial only).
E-104	Type not allowed	The RADix type was invalid
E-123	Path not found	The message used an invalid path command (USB/Serial only).
E-124	Data mismatch	The message contained data that did not match the expected format (USB/Serial only).
E-126	Too few or too many elements	The command requires more or less than the number of parameters actually supplied.
E-127	Change not allowed	An attempt was made to change a parameter that cannot be changed, or is currently read-only.
E-201	Data out of range	The message attempted to set a value that was outside the allowable range (USB/Serial only).
E-202	Invalid data type	When trying to parse the message, the data was in an invalid format (USB/Serial only).
E-204	Suffix not valid	An invalid number base suffix (radix) was encountered when parsing a number (USB/Serial only).
E-402	Sensor open, output turned off	A sensor open circuit was detected and the output was turned off.
E-403	Module open, output turned off	A Peltier module open circuit was detected and the output was turned off.
E-404	I limit, output turned off	A current limit was detected and the output was turned off.
E-405	V limit, output turned off	A voltage limit was detected and the output was turned off.
E-406	Thermistor resistance limit, output turned off	The thermistor resistance limit (high or low) was exceeded and the output was turned off.
E-407	Temperature limit, output turned off	The temperature limit (high or low) was exceeded and the output was turned off.
E-410	Temperature was out of tolerance, output turned off	The temperature went out of tolerance and the output was turned off.
E-415	Sensor short, output turned off	A sensor short circuit was detected and the output was turned off.
E-419	TEC not stable	The TEC is considered stable if the temperature has changed less than 0.02°C for more than 20 seconds.
E-433	Not a TEC	The TEC:CHAN command attempted to select a non-TEC channel
E-434	Ite limit exceeds cable rating	The cable plugged into the unit cannot carry the current as limited by

Error Code	Description	Cause
E-435	Mode change	the Ite Limit setting. Lower the Ite limit to the cables capacity, or use a higher capacity cable. A mode change occurred when the output was on, forcing the output off.
E-501	Interlock shutdown output	The interlock input was not shorted when the output was on.
E-504	Laser current limit disabled output.	The laser output was turned off because a current limit was detected and the corresponding bit in the OUTOFF register was set.
E-505	Laser voltage limit disabled output	The laser voltage exceeded the voltage limit and the output was turned off.
E-506	Laser photodiode current limit disabled output	The laser output was turned off because a photodiode current limit was detected and the corresponding bit in the OUTOFF register was set.
E-507	Laser photodiode power limit disabled output	The laser output was turned off because a photodiode power limit was detected and the corresponding bit in the OUTOFF register was set.
E-509	Laser short circuit disabled output	The laser output was turned off because a short condition was detected and the corresponding bit in the OUTOFF register was set.
E-510	Laser out of tolerance disabled output	The laser output was turned off because an out-of-tolerance condition was detected and the corresponding bit in the OUTOFF register was set.
E-511	Laser control error disabled output	A hardware control error was detected which forced a shutdown of the laser output.
E-512	Power failure	A power failure was detected.
E-514	Laser mode change disabled output	A change in the operating mode of the LaserSource while the output was on shutdown the output.
E-516	Incorrect configuration for calibration to start	The LaserSource was not configured properly, including the mode and output on state, to be able to start the desired calibration process.
E-517	Calibration must have the output on to start	The laser output must be on for the calibration process to start.
E-534	Po mode selected with PD Response set to zero	Attempted to select Po mode and PD Response was zero, or LaserSource was in Po mode and PD Response was set to zero.
E-535	Calibration cancelled	The active calibration process was cancelled.
E-536	Intermittent contact detected	The instrument detected an intermittent contact and shut down the laser output. If this is triggering falsely (such as in a noisy environment), the intermittent contact detection can be disabled in the main menu.
E-537	Thermal Limit Exceeded	The thermal load inside the instrument is too high, and the output was shutdown to protect the instrument.
W-800	External Voltage Sense Low	On instruments with external voltage sense, the external voltage measurement is much lower than the voltage at the output connector. This is only a warning, and does not indicate an actual problem.
W-801	Burst Mode, Hold Output	When in Io (Burst) mode, to turn the output on, the Output button must be held down for at least one second. If it is held down for less than one second, this warning message informs the user than the Output button press did not turn the output on.
E-997	Control error, cycle power	A hardware control error occurred, cycle power to resolve. If error continues to occur, contact factory.
E-998	Command not supported	A command was recognized but not supported by the LaserSource.
E-999	Non-specific error	A non-specific error was encountered.



373 Front Street, Suite B, Grover Beach, CA 93433

Tel: (805) 481-6684 Fax: (805) 481-6628

sales@arroyoinstruments.com

www.arroyoinstruments.com