



# **MIR Laser System**

## **Write Your Own Application**

- **Drivers**
- **Samples**
- **Command Reference**

Version: 1.0  
Date: 4/18/2017

---



# Contents

<b>Foreword</b>	<b>0</b>
<b>1 Write Your Own Application</b>	<b>2</b>
1.1 32 bit Version _____	3
1.2 64 bit Version _____	5
1.3 Command Reference _____	7
1.3.1 IEEE488.2 Common Commands _____	7
1.3.2 SCPI Command Reference _____	8
1.3.3 Status Reporting _____	14
1.3.4 Error Reporting _____	17
<b>2 Thorlabs Worldwide Contacts</b>	<b>21</b>

---

# 1 Write Your Own Application

In order to write your own application, you need a specific instrument driver and some tools for use in different programming environments. The driver and tools are being installed to your computer during software installation and cannot be found in the installation package.

In this section the location of drivers and files, required for programming in different environments, are given for installation under Windows VISTA, Windows 7, Windows 8.x and Windows 10 (32 and 64 bit).

In order to fully support 64 bit LabView version, the installation offers two installer versions:

- for Windows VISTA (32/64 bit), Windows 7 (32/64 bit), Windows 8.x (32/64 bit) and Windows 10 (32/64 bit): Install "TLTKL VXIpn Instrument Driver (32bit)"
- for Windows VISTA (64 bit), Windows 7 (64 bit), Windows 8.x (64 bit) and Windows 10 (64 bit): Install "TLTKL VXIpn Instrument Driver (64 bit)"

In other words, the 32 bit VXIpn driver works with both 32 and 64 bit operating systems, while the 64 bit driver requires a 64 bit operating system.

In the table below you will find a summary of what files you need for particular programming environments.

Programming environment	Necessary files
<b>C, C++, CVI</b>	*.h (header file) *.lib (static library)
<b>C#</b>	.net wrapper dll
<b>Visual Studio</b>	*.h (header file) *.lib (static library) or .net wrapper dll
<b>LabView</b>	*.fp (function panel) and VXIpn Instrument Driver Beside that, LabVIEW driver vi's are provided with the *.llb container file

## Note

All above environments require also the VXIpn Instrument Driver dll !

In the next sections the locations of above files are described in detail.

## 1.1 32 bit Version

### Note

According to the VPP6 (Rev6.1) Standard the installation of the 32 bit VXIpn driver includes both the WINNT and GWINNT frameworks.

### VXIpn Instrument driver:

C:\Program Files\IVI Foundation\VISA\WinNT\Bin\TLTKL\_32.dll

### Note

This instrument driver is required for all development environments!

### Header file

C:\Program Files\IVI Foundation\VISA\WinNT\include\TLTKL.h

### Static Library

C:\Program Files\IVI Foundation\VISA\WinNT\lib\msc\TLTKL\_32.lib

### Function Panel

C:\Program Files\IVI Foundation\VISA\WinNT\TLTKL\TLTKL.fp

### Online Help for VXIpn Instrument driver:

C:\Program Files\IVI Foundation\VISA\WinNT\TLTKL\Manual\TLTKL.html

### NI LabVIEW driver

The LabVIEW Driver is a 32 bit driver and compatible with 32bit NI-LabVIEW versions 8.5 and higher only.

C:\Program Files\National Instruments\LabVIEW xxxx\instr.lib\TLTKL...  
...\TLTKL.llb

(LabVIEW container file with driver vi's and an example. "LabVIEW xxxx" stands for actual LabVIEW installation folder.)

### .net wrapper dll

C:\Program Files\Microsoft.NET\Primary Interop Assemblies...  
...\Thorlabs.TLTKL\_32.interop.dll

C:\Program Files\IVI Foundation\VISA\VisaCom\...  
...\Primary Interop Assemblies\Thorlabs.TLTKL\_32.interop.dll

### Example for NI LabWindows/CVI (C)

#### Source file:

C:\Program Files\IVI Foundation\VISA\WinNT\TLTKL\Examples\C\...  
...sample.c

#### Executable sample demo:

C:\Program Files\IVI Foundation\VISA\WinNT\TLTKL\Examples\C\...  
...sample.exe

## **MS Visual Studio, .NET (C#)**

### **Solution file:**

```
C:\Program Files\IVI Foundation\VISA\WinNT\TLTKL\Examples...  
...\CSharp\Thorlabs.TLTKL.CSharpSample.sln
```

### **Project file:**

```
C:\Program Files\IVI Foundation\VISA\WinNT\TLTKL\Examples...  
...\CSharp\Thorlabs.TLTKL.CSharpSample\  
...Thorlabs.TLTKL.CSharpSample.csproj
```

### **Executable sample demo:**

```
C:\Program Files\IVI Foundation\VISA\WinNT\TLTKL\Examples...  
...\CSharp\Thorlabs.TLTKL.CSharpSample\bin\Release\  
...Thorlabs.TLTKL.CSharpSample.exe
```

(Select the correct type and device mode, e.g., TMC or DFU, and enter serial number, then connect)

## **Example for LabView**

```
C:\Program Files\National Instruments\LabVIEW xxxx\Instr.lib\TLTKL...  
...\TLTKL.llb
```

(LabVIEW container file with driver vi's and an example. "LabVIEW xxxx" stands for actual LabVIEW installation folder.)

## 1.2 64 bit Version

### Note

According to the VPP6 (Rev6.1) Standard the installation of the 64 bit VXIpn driver includes the WINNT, WIN64, GWINNT and GWIN64 frameworks. That means, that the 64 bit driver includes the 32 bit driver as well.

In case of a 64 bit operating system, 64bit drivers and applications are installed to

`"C:\Program Files"`

while the 32 bit files - to

`"C:\Program Files (x86)"`

Below are listed both installation locations, so far applicable.

### VXIpn Instrument driver:

`C:\Program Files (x86)\IVI Foundation\VISA\WinNT\Bin\TLTKL_32.dll`

`C:\Program Files\IVI Foundation\VISA\Win64\Bin\TLTKL_64.dll`

### Note

This instrument driver is required for all development environments!

### Header file

`C:\Program Files (x86)\IVI Foundation\VISA\WinNT\include\TLTKL.h`

`C:\Program Files\IVI Foundation\VISA\Win64\include\TLTKL.h`

### Static Library

`C:\Program Files (x86)\IVI Foundation\VISA\WinNT\lib\msc...  
...\TLTKL_32.lib`

`C:\Program Files\IVI Foundation\VISA\Win64\Lib_x64\msc\TLTKL_64.lib`

### Function Panel

`C:\Program Files (x86)\IVI Foundation\VISA\WinNT\TLTKL\TLTKL.fp`

### Online Help for VXIpn Instrument driver:

`C:\Program Files (x 86)\IVI Foundation\VISA\WinNT\TLTKL\Manual...  
...\TLTKL.html`

### NI LabVIEW driver

The LabVIEW Driver supports 32bit and 64bit NI-LabVIEW2009 and higher.

#### 32 bit NI-Labview version

`C:\Program Files (x86)\National Instruments\LabVIEW xxxx\instr.lib...  
...\TLTKL\TLTKL.llb`

#### 64 bit NI-Labview version

`C:\Program Files\National Instruments\LabVIEW xxxx\instr.lib...  
...\TLTKL\TLTKL.llb`

(LabVIEW container file with driver vi's and an example. "LabVIEW xxxx" stands for actual LabVIEW installation folder.)

### **.net wrapper dll**

C:\Program Files (x86)\Microsoft.NET\Primary Interop Assemblies...  
...\Thorlabs.TLTKL\_32.interop.dll

C:\Program Files (x86)\IVI foundation\VISA\VisaCom\...  
...\Primary Interop Assemblies\Thorlabs.TLTKL\_32.interop.dll

C:\Program Files\IVI foundation\VISA\VisaCom64\...  
...\Primary Interop Assemblies\Thorlabs.TLTKL\_64.interop.dll

### **Example for NI LabWindows/CVI (C)**

#### **Source file:**

C:\Program Files (x86)\IVI Foundation\VISA\WinNT\TLTKL\Examples\C\...  
...sample.c

#### **Executable sample demo:**

C:\Program Files (x86)\IVI Foundation\VISA\WinNT\TLTKL\Examples\C\...  
...sample.exe

### **MS Visual Studio, .NET (C#)**

#### **Solution file:**

C:\Program Files (x86)\IVI Foundation\VISA\WinNT\TLTKL\Examples...  
...\CSharp\Thorlabs.TLTKL.CSharpSample.sln

#### **Project file:**

C:\Program Files (x86)\IVI Foundation\VISA\WinNT\TLTKL\Examples...  
...\CSharp\Thorlabs.TLTKL.CSharpSample...  
...\Thorlabs.TLTKL.CSharpSample.csproj

#### **Executable sample demo:**

C:\Program Files (x86)\IVI Foundation\VISA\WinNT\TLTKL\Examples...  
...\CSharp\Thorlabs.TLTKL.CSharpSample...  
...\bin\Release\Thorlabs.TLTKL.CSharpSample.exe

(Select the correct type and device mode, e.g., TMC or DFU, and enter serial number, then connect)

### **Example for LabVIEW**

C:\Program Files\National Instruments\LabVIEW xxxx\Instr.lib\TLTKL...  
...\TLTKL.llb

(LabVIEW container file with driver vi's and an example. "LabVIEW xxxx" stands for actual LabVIEW installation folder.)



## 1.3 Command Reference

### 1.3.1 IEEE488.2 Common Commands

Common commands are device commands that are common to all devices according to the IEEE488.2 standard. These commands are designed and defined by this standard. Most of the commands are described in detail in this section. The following common commands associated with the status structure are covered in the “Status Structure” section: \*CLS, \*ESE, \*ESE?, \*ESR?, \*SRE,

#### Command summary

Mnemonic	Name	Description
*CLS	Clear status	Clears all event registers and Error Queue
*ESE <NRf>	Event enable command	Sets the Standard Event Enable Register
*ESE?	Event enable query	Returns the Standard Event Enable Register
*ESR?	Event status register query	Returns and clear the Standard Event Register
*IDN?	Identification query	Returns the unit's identification string
*OPC	Operation complete command	Sets the Operation Complete bit in the Standard Event Register
*OPC?	Operation complete query	Places a “1” into the output queue when all device operations have been completed
*RST	Reset command	Returns the unit to the *RST default condition
*SRE <NRf>	Service request enable command	Sets the Service Request Enable Register
*SRE?	Service request enable query	Returns the Service Request Enable Register
*STB?	Status byte query	Returns the Status Byte Register
*TST?	Self-test query	Performs the unit's self-test and returns the result.
*WAI	Wait-to-continue command	Waits until all previous commands are executed

#### Command reference

##### 1. \*IDN? – identification query - read identification code

The identification code includes the manufacturer, model code, serial number, and firmware revision levels and is sent in the following format: Thorlabs,MMM,SSS,X.X.X, where

MMM is the model code  
 SSS is the serial number  
 X.X.X is the instrument firmware revision level

##### 2. \*IDN2?

This command is the identification query of the connected laser head.

##### 3. \*OPC – operation complete - set OPC bit

##### 4. \*OPC? – operation complete query – places a “1” in output queue

When \*OPC is sent, the OPC bit in the Standard Event Register will 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. After the last pending command is executed, the OPC bit and/or an ASCII “1” is placed in the Output Queue.

### 5. \*RST – reset – return instrument to defaults

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

- Returns the instrument to default conditions
- Cancels all pending commands.
- Cancels response to any previously received \*OPC and \*OPC? commands.

### 6. \*TST? – self-test query – run self test and read result

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.

### 7. \*WAI – wait-to-continue – wait until previous commands are completed

The \*WAI command is not relevant for the instrument and thus, is not used. It was included only for conformance with IEEE488.2.

## 1.3.2 SCPI Command Reference

### SYSTEM subsystem commands

Command	Description	SCPI
SYSTEM	Path to SYSTEM subsystem.	<input checked="" type="checkbox"/>
:BEEPer		<input checked="" type="checkbox"/>
[:IMMediate]	Issues an audible signal	<input checked="" type="checkbox"/>
:STATe {ON 1 OFF 0}	Activates/deactivates the beeper	<input checked="" type="checkbox"/>
:STATe?	Returns the state of the beeper	<input checked="" type="checkbox"/>
:VOLume <value>	Sets the beeper volume	<input checked="" type="checkbox"/>
:VOLume?	Returns the beeper volume	<input checked="" type="checkbox"/>
:ERRor		<input checked="" type="checkbox"/>
[:NEXT]?	Returns the latest error code and message	<input checked="" type="checkbox"/>
:MOUNt		
[:TYPE]?	Returns the mount type (<NR1>,description)	
:VERSion?	Returns level of SCPI standard (1999.0)	<input checked="" type="checkbox"/>
:USED {ON 1 OFF 0}	Sets the used by remote state	
:USED?	Returns the used by remote state	

### DISPlay subsystem commands

Command	Description	SCPI
DISPlay	Path to DISPlay subsystem.	<input checked="" type="checkbox"/>
:BRIGHtness <value>	Sets the display brightness	<input checked="" type="checkbox"/>
:BRIGHtness?	Returns the display brightness value	<input checked="" type="checkbox"/>
:CALibratIon[:TOUCH][:INITiate]	Initiates Touchscreen calibration	
:FADeout		
[:STATe] {ON 1 OFF 0}	Activates/deactivates automatic dimming	
[:STATe]?	Returns the state of automatic dimming	

## STATus subsystem commands

Command	Description	SCPI
<b>STATus</b>		<input checked="" type="checkbox"/>
<b>:MEASurement</b>	Path to control measurement event registers	
[:EVENT]?	Returns the event register	
:CONDition?	Returns the condition register	
:PTRansition <value>	Sets the positive transition filter	
:PTRansition?	Returns the positive transition filter	
:NTRansition <value>	Sets the negative transition filter	
:NTRansition?	Returns the negative transition filter	
:ENABle <value>	Sets the enable register	
:ENABle?	Returns the enable register	
<b>:OPERation</b>	Path to control operation event registers	<input checked="" type="checkbox"/>
[:EVENT]?	Returns the event register	<input checked="" type="checkbox"/>
:CONDition?	Returns the condition register	<input checked="" type="checkbox"/>
:PTRansition <value>	Sets the positive transition filter	<input checked="" type="checkbox"/>
:PTRansition?	Returns the positive transition filter	<input checked="" type="checkbox"/>
:NTRansition <value>	Sets the negative transition filter	
:NTRansition?	Returns the negative transition filter	<input checked="" type="checkbox"/>
:ENABle <value>	Sets the enable register	<input checked="" type="checkbox"/>
:ENABle?	Returns the enable register	<input checked="" type="checkbox"/>
<b>:QUESTionable</b>	Path to control questionable event registers	<input checked="" type="checkbox"/>
[:EVENT]?	Returns the event register	<input checked="" type="checkbox"/>
:CONDition?	Returns the condition register	<input checked="" type="checkbox"/>
:PTRansition <value>	Sets the positive transition filter	<input checked="" type="checkbox"/>
:PTRansition?	Returns the positive transition filter	<input checked="" type="checkbox"/>
:NTRansition <value>	Sets the negative transition filter	<input checked="" type="checkbox"/>
:NTRansition?	Returns the negative transition filter	<input checked="" type="checkbox"/>
:ENABle <value>	Sets the enable register	<input checked="" type="checkbox"/>
:ENABle?	Returns the enable register	<input checked="" type="checkbox"/>
<b>:AUXiliary</b>	Path to control auxiliary event registers	
[:EVENT]?	Returns the event register	
:CONDition?	Returns the condition register	
:PTRansition <value>	Sets the positive transition filter	
:PTRansition?	Returns the positive transition filter	
:NTRansition <value>	Sets the negative transition filter	
:NTRansition?	Returns the negative transition filter	
:ENABle <value>	Sets the enable register	
:ENABle?	Returns the enable register	
<b>:PRESet</b>	Return status registers to default states.	<input checked="" type="checkbox"/>

**LD output subsystem commands**

Command	Description	SCPI
<b>OUTPut[1]</b>	<b>Path to LD output</b>	<input checked="" type="checkbox"/>
[:STATE] {ON 1 OFF 0}	Enables (ON) or disables (OFF) LD output	<input checked="" type="checkbox"/>
[:STATE]?	Returns output state	<input checked="" type="checkbox"/>
<b>:PON</b>	<b>Path to LD power ON</b>	
:DElay {MIN MAX DEF <seconds>}	Sets the LD output power-on delay	
:DElay? [{MIN MAX DEF}]	Returns the LD output power-on delay setting	
:CONDition?	Returns the output condition (query only, 1 0)	
<b>:FILTer[:LPASs]</b>	<b>Path to LD output filter</b>	<input checked="" type="checkbox"/>
[:STATE] {ON 1 OFF 0}	Enables/disables LD output low pass filter	<input checked="" type="checkbox"/>
[:STATE]?	Returns output filter state	<input checked="" type="checkbox"/>
<b>:PROtEction</b>	<b>Path to LD output protection</b>	<input checked="" type="checkbox"/>
:INTLock[:TRIPped]?	Returns interlock circuit protection tripped	
:KEYLock[:TRIPped]?	Returns key lock protection tripped	
:OTEMperature[:TRIPped]?	Returns over temperature protection tripped	
:CONNecTion[:TRIPped]?	Returns connection failure protection tripped	
<b>:TEMPerature</b>		
:MODE {OFF PROtEction ENABle}	Sets temperature protection mode	
:MODE?	Returns temperature protection mode	
{:TRIPped}?	Returns protection tripped	

**LD current sensing subsystem commands**

Command	Description	SCPI
<b>SENSE3</b>	<b>Path to laser diode current sensing</b>	<input checked="" type="checkbox"/>
[:CURRENT] [:DC]		<input checked="" type="checkbox"/>
[:DATA]? [{MIN MAX}]	Returns the measured LD current	
:FAN:SPEED? [{MIN MAX}]	Returns the speed of the laser head fan	

**LD voltage sensing subsystem commands**

Command	Description	SCPI
<b>SENSE4</b>	<b>Path to laser diode voltage sensing</b>	<input checked="" type="checkbox"/>
[:VOLTage] [:DC]		<input checked="" type="checkbox"/>
[:DATA]? [{MIN MAX}]	Returns the measured LD voltage	

**LD source subsystem commands**

Command	Description	SCPI
<b>SOURce[1]</b>	<b>Path to Laser output</b>	<input checked="" type="checkbox"/>
[:CURRENT]	<b>Path to Laser output current</b>	<input checked="" type="checkbox"/>
:LIMit		<input checked="" type="checkbox"/>
[:AMPLitude] {MIN MAX <amps>}	Sets limit current value	<input checked="" type="checkbox"/>
[:AMPLitude]? [{MIN MAX}]	Returns limit current value	<input checked="" type="checkbox"/>
:TRIPped?	Returns limit detection tripped	
[:LEVel] [:IMMediate]		<input checked="" type="checkbox"/>
[:AMPLitude] {MIN MAX <amps>}	Sets LD current setpoint value	<input checked="" type="checkbox"/>
[:AMPLitude]? [{MIN MAX}]	Returns LD current setpoint value	<input checked="" type="checkbox"/>
<b>:VOLTage</b>	<b>Path to Laser output voltage</b>	<input checked="" type="checkbox"/>
[:LEVel]		<input checked="" type="checkbox"/>
[:IMMediate]		<input checked="" type="checkbox"/>
[:AMPLitude]? [{MIN MAX}]	Returns LD voltage setpoint value	<input checked="" type="checkbox"/>
:DIODE[:CURRENT] [:IMMediate]	Sets LD power via photodiode current	
[:AMPLitude] {MIN MAX <amps>}	Sets photodiode current setpoint	
[:AMPLitude]? [{MIN MAX}]	Returns the photodiode current setpoint	

Command	Description	SCPI
<b>:AM</b>	<b>Path to Laser output modulation</b>	<input checked="" type="checkbox"/>
<b>:INTernal</b>		<input checked="" type="checkbox"/>
[:STATE] {ON 1 OFF 0}	Enable (ON) or disable (OFF) internal mod.	
[:STATE]?	Returns internal modulation state	
:AMPLitude {MIN MAX <amps>}	Sets LD internal modulation amplitude	
:AMPLitude? [{MIN MAX}]	Returns LD internal modulation amplitude	
:FREQuency {MIN MAX <hertz>}	Sets LD internal modulation frequency	<input checked="" type="checkbox"/>
:FREQuency? [{MIN MAX}]	Returns LD internal modulation frequency	<input checked="" type="checkbox"/>
:FUNCTion[:SHAPE]	Sets LD internal modulation shape	
{SINusoid 1 TRIangle 2 SQUare 3}		
:FUNCTion[:SHAPE]?	Returns LD internal modulation shape	
<b>:EXTernal</b>		
[:STATE] {ON 1 OFF 0}	Enable (ON) or disable (OFF) external mod.	
[:STATE]?	Returns external modulation state	
:RANGE {LOW 0 HIGH 2}	Sets the external modulation range	
:RANGE?	Returns the external modulation range	
<b>:TUNE:EXT</b>	<b>Path to Laser output tuning</b>	
[:STATE] {ON 1 OFF 0}	Enable (ON) or disable (OFF) external tuning	
[:STATE]?	Returns external tuning state	

### TEC output subsystem commands

Command	Description	SCPI
<b>OUTPut2</b>	<b>Path to TEC output</b>	<input checked="" type="checkbox"/>
[:STATE] {ON 1 OFF 0}	Enable (ON) or disable (OFF) TEC output	<input checked="" type="checkbox"/>
[:STATE]?	Returns output state	<input checked="" type="checkbox"/>
<b>:PON</b>	<b>Path to TEC output power on</b>	
[:STATE] {ON 1 OFF 0}	Sets the power-on TEC output state	
[:STATE]?	Returns power-on TEC output state setting	
<b>:PROtection</b>	<b>Path to TEC output protection</b>	<input checked="" type="checkbox"/>
:TRANsducer[:TRIPped]?	Returns temperature transducer failure tripped	
:OTEMperature[:TRIPped]?	Returns over temperature protection tripped	
:CONNECTION[:TRIPped]?	Returns connection failure protection tripped	

### TEC driver source subsystem commands

Command	Description	SCPI
<b>SOURce2</b>	<b>Path to TEC output</b>	<input checked="" type="checkbox"/>
<b>:CURRent</b>	<b>Path to TEC output current</b>	<input checked="" type="checkbox"/>
<b>:LIMit</b>		<input checked="" type="checkbox"/>
[:AMPLitude] {MIN MAX <amps>}	Sets limit current value	<input checked="" type="checkbox"/>
[:AMPLitude]? [{MIN MAX}]	Returns limit current value	<input checked="" type="checkbox"/>
:TRIPped?	Returns limit detection tripped	<input checked="" type="checkbox"/>
:DATA? [{MIN MAX}]	Returns the TEC current value	
[:TEMPerature]	<b>Path to TEC output temperature</b>	<input checked="" type="checkbox"/>
[:SPOint] {MIN MAX DEF <temp>}	Sets temperature setpoint	<input checked="" type="checkbox"/>
[:SPOint]? [{MIN MAX DEF}]	Returns temperature setpoint	<input checked="" type="checkbox"/>
<b>:LIMit</b>		
[:UPPer] {MIN MAX <temp>}	Sets settable temperature high limit	
[:UPPer]? [{MIN MAX}]	Returns settable temperature high limit	
:LOWer {MIN MAX <temp>}	Sets settable temperature low limit	
:LOWer? [{MIN MAX}]	Returns settable temperature low limit	

## Temperature sensing subsystem commands

Command	Description	SCPI
SENSe2		<input checked="" type="checkbox"/>
[:TEMPerature]		
:DATA? [{MIN MAX}]	Returns the temperature value	
:PROTection		
:DELay {MIN MAX DEF <sec>}	Sets protection delay	
:DELay? [{MIN MAX DEF}]	Returns protection delay	
:WINDow[:AMPLitude]	Sets temperature window amplitude	
{MIN MAX DEF <temp>}		
:WINDow[:AMPLitude]?	Returns temperature window amplitude	
[{MIN MAX DEF}]		
[:TRIPped]?	Returns protection tripped	
:RESistance		
[:DATA]? [{MIN MAX}]	Returns the resistance value	
:FAN:SPEEd? [{MIN MAX}]	Returns the controller's fan speed	

## Laser head subsystem commands

Command	Description	SCPI
SOURce3		<input checked="" type="checkbox"/>
:INTPol		
:TYPE {NONE 0 WLEN 1 POW 3}	Sets the interpolation type	
:TYPE?	Returns the interpolation type	
:WAVElength <value>	Sets the laser wavelength	
:WAVElength? [{MIN MAX}]	Returns the laser wavelength	
:WAVElength:NOMinal?	Returns the nominal laser wavelength	
:POWer <watt>	Sets the laser output power	
:POWer? [{MIN MAX}]	Returns the laser output power	
:POWer:NOMinal?	Returns the nominal laser output power	
:CURRent:LIMit?:MAXimum?	Returns the laser current limit	
:VOLTage:LIMit:MAXimum?	Returns the laser voltage limit	

## UNIT subsystem commands

Command	Description	SCPI
UNIT		<input checked="" type="checkbox"/>
:TEMPerature	Sets the temperature unit	<input checked="" type="checkbox"/>
{C CEL CELSius F FAR FAHReinheit K KELVin}		
:TEMPerature?	Returns the temperature unit	<input checked="" type="checkbox"/>
:WAVElength {1 NM 2 WNUM}	Sets the WL unit to wavelength [nm] or wave number	
:WAVElength?	Returns the WL unit	

## CALibration subsystem commands

Command	Description	SCPI
CALibration		<input checked="" type="checkbox"/>
:STRing?	Returns the calibration string of the controller	
CALibration2		
:STRing?	Returns the calibration string of the laser head	

**MANufacturer subsystem commands**

Command	Description	SCPI
<b>MANufacturer</b> :STRing?	Returns the manufacturer string of the controller	
<b>MANufacturer2</b> :STRing?	Returns the manufacturer string of the laser head	

**Measurement commands**

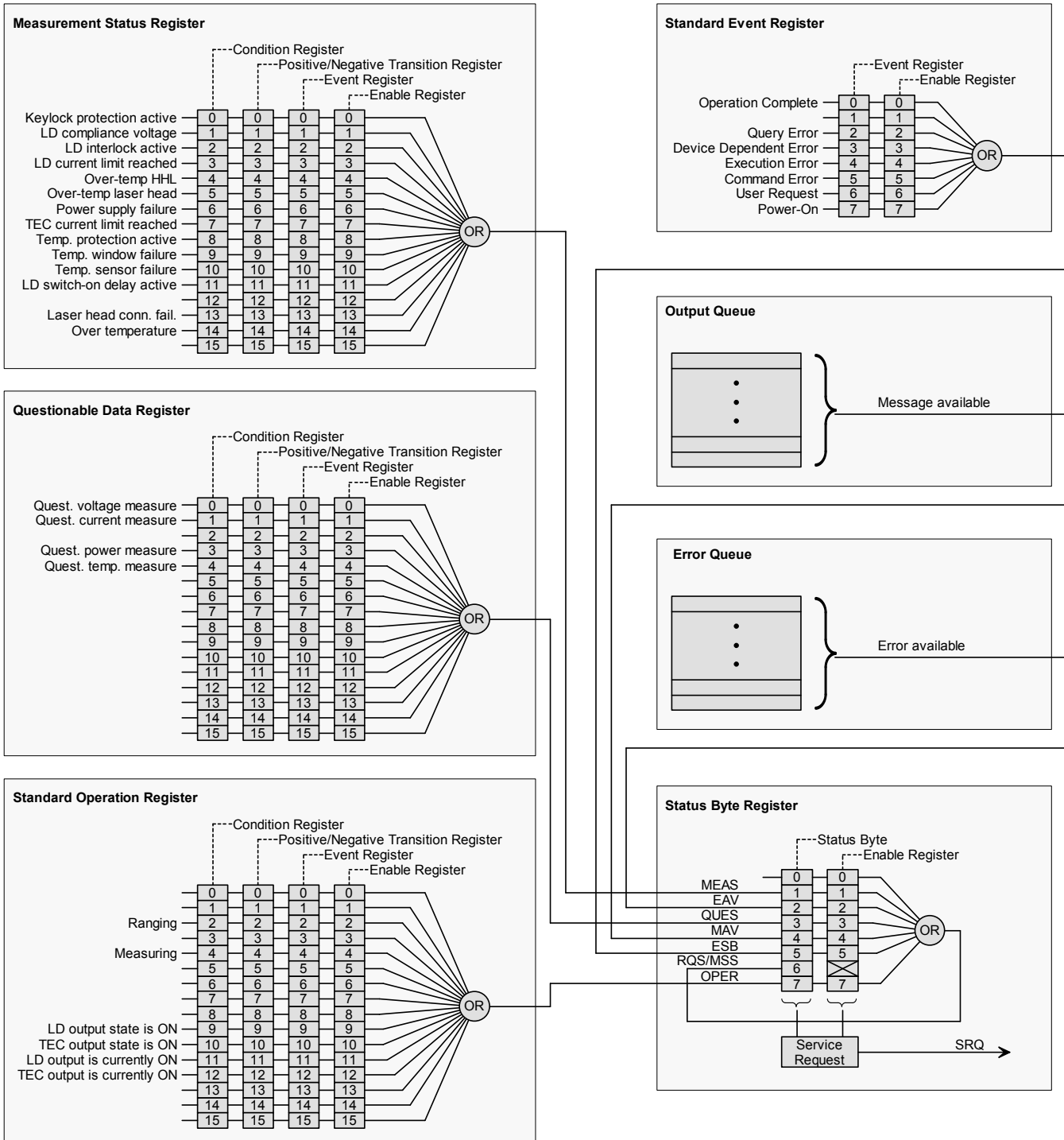
Command	Description	SCPI
<b>ABORT</b>	Aborts current measurement	<input checked="" type="checkbox"/>
<b>CONFigure</b> [:SCALar]		<input checked="" type="checkbox"/>
:CURRent[1] [:DC]	Configures instrument LD current measurement	<input checked="" type="checkbox"/>
:VOLTagE[1] [:DC]	Configures instrument LD voltage measurement	<input checked="" type="checkbox"/>
:TEMPerature	Configures instrument for temperature measurement	<input checked="" type="checkbox"/>
:RESistance	Configures instrument NTC resistance measurement	<input checked="" type="checkbox"/>
<b>CONFigure?</b>	Query configuration	<input checked="" type="checkbox"/>
<b>INITiate</b> [:IMMediate]	Starts measurement	<input checked="" type="checkbox"/>
<b>FETCH?</b>	Returns last measurement data	<input checked="" type="checkbox"/>
<b>FETCH</b>		
:CURRent[1] [:DC]?	Return last LD current measurement	<input checked="" type="checkbox"/>
:VOLTagE[1] [:DC]?	Return last LD voltage measurement	<input checked="" type="checkbox"/>
:TEMPerature?	Return last temperature measurement	<input checked="" type="checkbox"/>
:RESistance?	Return last NTC resistance measurement	<input checked="" type="checkbox"/>
<b>READ?</b>	Starts new measurement (as configured) and read data	<input checked="" type="checkbox"/>
<b>MEASure</b> [:SCALar]		
:CURRent[1] [:DC]?	Perform LD current measurement	<input checked="" type="checkbox"/>
:VOLTagE[1] [:DC]?	Perform LD voltage measurement	<input checked="" type="checkbox"/>
:TEMPerature?	Perform LD temperature measurement	<input checked="" type="checkbox"/>
:HHL?	Perform laser head High-Heat Load temperature measurement	
:T2?	Perform laser head transistor2 temperature measurement	
:SHUNt?	Perform laser head shunt temperature measurement	
:RESistance?	Perform NTC resistance measurement	<input checked="" type="checkbox"/>

**MEMory subsystem commands**

Command	Description	SCPI
<b>MEMory</b> :SAVE [IMMediate]	Stores current device setup to flash memory	<input checked="" type="checkbox"/>

### 1.3.3 Status Reporting

The figure below gives an overview of the device's status reporting structure. See also section [STATUS subsystem commands](#) for a detailed description of the related commands and their syntax.





## Status Byte Register

The Status Byte Register gives a summary of all underlying status structures. See also IEEE488.2-1992-§11.2.

Bit #	Mnemonic	Description
7	<b>OPER</b>	Standard Operation Status Structure Summary Bit
6	<b>RQS/MSS</b>	Request Service / Master Summary Status
5	<b>ESB</b>	Standard Event Status Bit
4	<b>MAV</b>	Message Available. There is response data available for readout
3	<b>QUES</b>	Questionable Status Structure Summary Bit
2	<b>EAV</b>	Error Available. There is at least one error in the error queue.
1	<b>MEAS</b>	Measurement Status Structure Summary Bit
0		reserved, read as 0

## Standard Event Status Structure

The Standard Event Status Structure is described in IEEE488.2-1992-§11.5.

## Standard Operation Register

The Standard Operation Status Structure is described in SCPI1999.0-Vol1-§9.3. In addition bit 8 to 12 are used as output state/on indicators.

Bit #	Mnemonic	Description
15..13		See SCPI1999.0-Vol1-§9.3
12	<b>TECON</b>	TEC output is currently ON
11	<b>LDON</b>	LD output is currently ON
10	<b>TECST</b>	TEC output state is ON
9	<b>LDST</b>	LD output state is ON
8		reserved, read as 0
7..0		See SCPI1999.0-Vol1-§9.3

## Questionable Data Register

The Questionable Data Status Structure is described in SCPI1999.0 Vol1 §9.4.

**Measurement Status Register**

The Measurement Status Register Status Byte Register reports device operation and measurement states.

Bit #	Description
15	reserved, read as 0
14	Over temperature (Instrument is too hot)
13	Laser head connection failure (Laser head missing or unknown laser head detected)
12	reserved, read as 0
11	LD switch-on delay active
10	Temperature sensor failure.
9	Temperature window failure.
8	Temperature protection is active.
7	TEC current limit reached
6	Power supply failure
5	Over-temperature of the laser head
4	Over-temperature of the HHL high-heat load
3	LD current limit reached
2	LD interlock is active
1	LD output compliance voltage reached
0	Keylock protection is active

### 1.3.4 Error Reporting

The device stores errors in a queue containing up to 10 entries. The error queue may be read out by the ``SYSTEM:ERROR[:NEXT]?'` command. The following table lists all error numbers and the according descriptive messages. Note: negative numbers are defined by SCPI while positive error numbers are device dependent.

Error	Description
0	No error
1	The error couldn't be specified more precisely
3	Device temperature too high
4	General GUI error
5	Authentication required for operation
6	Authentication process failed
7	Operation is not allowed in service mode
8	Operation is allowed in service mode only
11	Not allowed to change value in REMOTE mode
12	Not allowed to modify value in REMOTE mode
13	Not allowed to switch outputs in REMOTE mode
14	Laser head missing
15	Power supply error
17	Over-temperature laser head
20	Operation not allowed while LD output is on
22	INTERLOCK circuit is open
23	KEYLOCK is active
24	Operation not allowed because of a 'OPEN CIRCUIT' condition
25	TEC is off
26	TEC goes off
27	Temperature Protection is active
28	NTC failure - LD output cannot switch on
29	Power supply laser head failure
30	Operation not allowed while interpolation is on
31	Operation not allowed while wavelength interpolation is off
32	Operation not allowed while wavenumber interpolation is off
33	Operation not allowed while power interpolation is off
34	Modulation amplitude is decreased by frequency above specs for sinusoidal shape without NR filter
35	Modulation frequency reduced to maximum for triangle shape without NR filter
36	Modulation frequency reduced to maximum for square shape without NR filter
37	Modulation frequency reduced to maximum for sinusoidal shape without NR filter
38	Modulation frequency reduced to maximum for triangle shape with NR filter
39	Modulation frequency reduced to maximum for square shape with NR filter

Error	Description
<b>50</b>	Operation not allowed while TEC output is on
<b>51</b>	Wrong operating mode for this operation
<b>52</b>	Operation not allowed while a procedure is running
<b>53</b>	Operation not allowed because of a 'SENSOR FAILURE' condition
<b>93</b>	Erroneous connection to thermistor A/D converter
<b>100</b>	I <sup>2</sup> C wires stuck - bus 0
<b>101</b>	Illegal START/STOP condition - bus 0
<b>102</b>	Slave address not acknowledged (Not a valid bus address?) - bus 0
<b>103</b>	Incomplete write operation (Slave rejected to receive all data in the buffer) - bus 0
<b>104</b>	Arbitration lost - bus 0
<b>110</b>	I <sup>2</sup> C wires stuck - bus 1
<b>111</b>	Illegal START/STOP condition - bus 1
<b>112</b>	Slave address not acknowledged (Not a valid bus address?) - bus 1
<b>113</b>	Incomplete write operation (Slave rejected to receive all data in the buffer) - bus 1
<b>114</b>	Arbitration lost - bus 1
<b>120</b>	I <sup>2</sup> C wires stuck - bus 2
<b>121</b>	Illegal START/STOP condition - bus 2
<b>122</b>	Slave address not acknowledged (Not a valid bus address?) - bus 2
<b>123</b>	Incomplete write operation (Slave rejected to receive all data in the buffer) - bus 2
<b>124</b>	Arbitration lost - bus 2
<b>130</b>	EEPROM Timeout
<b>131</b>	EEPROM Check-sum error
<b>132</b>	EEPROM memory address overflow
<b>133</b>	EEPROM memory not supported
<b>134</b>	EEPROM memory not detected
<b>135</b>	EEPROM asynchronous transfer already running
<b>150</b>	MLSC fan controller not responding
<b>151</b>	MLSC fan not spinning
<b>152</b>	MLSC heat sink temperature sensor failure
<b>153</b>	Laser head heat sink temperature sensor failure
<b>154</b>	MLSC over-temperature signal failure
<b>155</b>	Laser head fan controller not responding
<b>156</b>	Laser head fan not spinning

<b>Error</b>	<b>Description</b>
<b>160</b>	External power supply failure
<b>161</b>	Internal analog power supply failure
<b>170</b>	RAM device failure
<b>171</b>	RAM address failure
<b>172</b>	RAM data bus failure
<b>181</b>	Touch controller interrupt signal failure
<b>182</b>	Touch controller command error
<b>183</b>	Touch controller unrecognized command
<b>184</b>	Touch controller unrecognized header
<b>185</b>	Touch controller command time-out
<b>186</b>	Touch panel is not calibrated
<b>187</b>	Touch calibration canceled
<b>188</b>	Touch calibration already running
<b>189</b>	Touch calibration is not running
<b>190</b>	Touch calibration point is out of bounds
<b>200</b>	GUI value not editable
<b>210</b>	Numeric value error
<b>211</b>	Numeric value is at minimum
<b>212</b>	Numeric value is at maximum
<b>213</b>	Entry digit is at minimum
<b>214</b>	Entry digit is at maximum
<b>220</b>	Selection limit reached
<b>230</b>	Value is out of range
<b>251</b>	Values from memory button have been corrected
<b>252</b>	Value not stored in memory button
<b>253</b>	No value stored in memory button
<b>254</b>	Values stored to memory button (hint only)
<b>-100</b>	General command error
<b>-102</b>	Syntax error
<b>-108</b>	Parameter not allowed
<b>-113</b>	Undefined header (Unknown command)
<b>-115</b>	Unexpected number of parameters

Error	Description
<b>-120</b>	Numeric data error
<b>-130</b>	Suffix error
<b>-131</b>	Invalid suffix
<b>-150</b>	String data error
<b>-151</b>	Invalid string data
<b>-220</b>	Parameter error
<b>-221</b>	Settings conflict
<b>-222</b>	Data out of range
<b>-224</b>	Parameter value illegal
<b>-230</b>	Data corrupt or stale
<b>-240</b>	Hardware error
<b>-310</b>	System error
<b>-311</b>	Memory error
<b>-313</b>	Calibration memory lost
<b>-314</b>	Save/recall memory lost
<b>-315</b>	Configuration memory lost
<b>-321</b>	Out of memory
<b>-330</b>	Self-test failed
<b>-350</b>	Queue overflow
<b>-363</b>	Input buffer overrun
<b>-410</b>	Query INTERRUPTED

## 2 Thorlabs Worldwide Contacts

### USA, Canada, and South America

Thorlabs, Inc.  
56 Sparta Avenue  
Newton, NJ 07860  
USA  
Tel: 973-579-7227  
Fax: 973-300-3600  
[www.thorlabs.com](http://www.thorlabs.com)  
[www.thorlabs.us](http://www.thorlabs.us) (West Coast)  
Email: [sales@thorlabs.com](mailto:sales@thorlabs.com)  
Support: [techsupport@thorlabs.com](mailto:techsupport@thorlabs.com)

### Europe

Thorlabs GmbH  
Hans-Böckler-Str. 6  
85221 Dachau  
Germany  
Tel: +49-8131-5956-0  
Fax: +49-8131-5956-99  
[www.thorlabs.de](http://www.thorlabs.de)  
Email: [europe@thorlabs.com](mailto:europe@thorlabs.com)

### France

Thorlabs SAS  
109, rue des Côtes  
78600 Maisons-Laffitte  
France  
Tel: +33-970 444 844  
Fax: +33-811 38 17 48  
[www.thorlabs.com](http://www.thorlabs.com)  
Email: [sales.fr@thorlabs.com](mailto:sales.fr@thorlabs.com)

### Japan

Thorlabs Japan, Inc.  
Higashi Ikebukuro  
Q Building 2nd Floor 2-23-2  
Toshima-ku, Tokyo 170-0013  
Japan  
Tel: +81-3-5979-8889  
Fax: +81-3-5979-7285  
[www.thorlabs.jp](http://www.thorlabs.jp)  
Email: [sales@thorlabs.jp](mailto:sales@thorlabs.jp)

### UK and Ireland

Thorlabs Ltd.  
1 Saint Thomas Place, Ely  
Cambridgeshire CB7 4EX  
United Kingdom  
Tel: +44-1353-654440  
Fax: +44-1353-654444  
[www.thorlabs.com](http://www.thorlabs.com)  
Email: [sales.uk@thorlabs.com](mailto:sales.uk@thorlabs.com)  
Support: [techsupport.uk@thorlabs.com](mailto:techsupport.uk@thorlabs.com)

### Scandinavia

Thorlabs Sweden AB  
Bergfotsgatan 7  
431 35 Mölndal  
Sweden  
Tel: +46-31-733-30-00  
Fax: +46-31-703-40-45  
[www.thorlabs.com](http://www.thorlabs.com)  
Email: [scandinavia@thorlabs.com](mailto:scandinavia@thorlabs.com)

### Brazil

Thorlabs Vendas de Fotônicos Ltda.  
Rua Riachuelo, 171  
São Carlos, SP 13560-110  
Brazil  
Tel: +55-16-3413 7062  
Fax: +55-16-3413 7064  
[www.thorlabs.com](http://www.thorlabs.com)  
Email: [brasil@thorlabs.com](mailto:brasil@thorlabs.com)

### China

Thorlabs China  
Room A101, No. 100  
Lane 2891, South Qilianshan Road  
Putuo District  
Shanghai 200331  
China  
Tel: +86-21-60561122  
Fax: +86-21-32513480  
[www.thorlabs.com](http://www.thorlabs.com)  
Email: [chinasales@thorlabs.com](mailto:chinasales@thorlabs.com)



**THORLABS**

[www.thorlabs.com](http://www.thorlabs.com)

---