



TANGO
Device
Server

Class for the IPG Laser YLR 100 SM User's Guide

IPGLaserYLR100SM Class

**Revision: - Author: tnunez
Implemented in C++**

Introduction:

Class providing the interfacier for the IPG Laser YLR 100 SM.

Class Identification:

- **Contact** : at mail.desy.de - tnunez
- **Class Family** : Instrumentation
- **Platform** : Unix Like
- **Bus** : Socket
- **Manufacturer** : IPG Laser
- **Reference** : YLR-100-AC

Class Inheritance:

- Tango::Device_4Impl
 - IPGLaserYLR100SM

Properties:

Device Properties		
Property name	Property type	Description
SocketDS	Tango::DEV_STRING	Name of the IPGLaserYLRCtrl Tango device making the socket connection.
SimulationMode	Tango::DEV_LONG	Not used.

Device Properties Default Values:

Property Name	Default Values
SocketDS	No default value
SimulationMode	0

There is no Class properties.

States:

States	
Names	Descriptions
ON	
FAULT	

Attributes:

Scalar Attributes			
Attribute name	Data Type	R/W Type	Expert
Current: Sets and reads pump current.	DEV_DOUBLE	READ_WRITE	No
OutputPower: Output power in watts.	DEV_DOUBLE	READ_WRITE	No
Temperature	DEV_FLOAT	READ	No
Emission: 0 -> off, 1 -> on	DEV_LONG	READ	No
LaserStatus: Laser status as integer	DEV_LONG	READ	No
ExternalMode: Set External mode (independent of the value you write).	DEV_LONG	READ_WRITE	No
RedLaser: Set the red laser emission: 0->off, 1->on	DEV_LONG	READ_WRITE	No

Commands:

More Details on commands....

Device Commands for Operator Level		
Command name	Argument In	Argument Out
Init	DEV_VOID	DEV_VOID
State	DEV_VOID	DEV_STATE
Status	DEV_VOID	CONST_DEV_STRING

1 - Init

- **Description:** This commands re-initialise a device keeping the same network connection.
After an Init command executed on a device, it is not necessary for client to re-connect to the device.
This command first calls the device *delete_device()* method and then execute its *init_device()* method.
For C++ device server, all the memory allocated in the *nit_device()* method must be freed in the *delete_device()* method.
The language device desctructor automatically calls the *delete_device()* method.
- **Argin:**
DEV_VOID : none.
- **Argout:**
DEV_VOID : none.
- **Command allowed for:**

- Tango::ON
- Tango::FAULT

2 - State

- **Description:** This command gets the device state (stored in its *device_state* data member) and returns it to the caller.
- **Argin:**
DEV_VOID : none.
- **Argout:**
DEV_STATE : State Code
- **Command allowed for:**
 - Tango::ON
 - Tango::FAULT

3 - Status

- **Description:** This command gets the device status (stored in its *device_status* data member) and returns it to the caller.
- **Argin:**
DEV_VOID : none.
- **Argout:**
CONST_DEV_STRING : Status description
- **Command allowed for:**
 - Tango::ON
 - Tango::FAULT

TANGO is an open source project hosted by :


Core and Tools : CVS repository on tango-cs project
 Device Servers : CVS repository on tango-ds project