



TANGO
Device
Server

Focusing Mirror P02 User's Guide

FocusingMirrorP02 Class

Revision: - Author: tnunez
Implemented in C++

Introduction:

Class for handling focusing mirrors in p02

Class Identification:

- **Contact** : at mail.desy.de - tnunez
- **Class Family** : Motion
- **Platform** : Unix Like
- **Bus** : Not Applicable

Class Inheritance:

- Tango::Device_4Impl
 - FocusingMirrorP02

Properties:

Device Properties		
Property name	Property type	Description
UpstreamTiltMotor	Tango::DEV_STRING	Tango device name for the upstream tilt motor.
DownstreamTiltMotor	Tango::DEV_STRING	Tango device name for the downstream tilt motor.
UpstreamBentMotor	Tango::DEV_STRING	Tango device name for the upstream bent motor.
DownstreamBentMotor	Tango::DEV_STRING	Tango device name for the downstream bent motor.

Device Properties Default Values:

Property Name	Default Values
UpstreamTiltMotor	No default value
DownstreamTiltMotor	No default value
UpstreamBentMotor	No default value
DownstreamBentMotor	No default value

There is no Class properties.

States:

States	
Names	Descriptions
ON	
MOVING	
FAULT	

Attributes:

Scalar Attributes			
Attribute name	Data Type	R/W Type	Expert
ZPos	DEV_DOUBLE	READ_WRITE	No
Tilt	DEV_DOUBLE	READ_WRITE	No
Curvature	DEV_DOUBLE	READ_WRITE	No
Ellipticity	DEV_DOUBLE	READ_WRITE	No
MirrorName	DEV_STRING	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
StopMove	DEV_VOID	DEV_VOID

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::MOVING

- 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::MOVING
 - 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::MOVING
 - Tango::FAULT

4 - StopMove

- **Description:** Stop fthe four connected motors.
- **Argin:**
DEV_VOID :
- **Argout:**
DEV_VOID :
- **Command allowed for:**
 - Tango::ON

- Tango::MOVING
- Tango::FAULT

TANGO is an open source project hosted by : SOURCEFORGE.NET®	Core and Tools : CVS repository on tango-cs project Device Servers : CVS repository on tango-ds project
--	--