


TANGO Device Server

GalilCoupledAxes User's Guide

GalilCoupledAxes Class

Revision: release_2_3_10 - Author: leclercq
Implemented in C++

Introduction:

This class is used to control coupled axes by controlling only the master axis. All the master-slave axis configuration are done by motorists in micro-code itself. Special commands if needed are accessed through MicrocodeDataViewer DS First need : a detector table with 2 motors

Class Inheritance:

- Tango::Device_3Impl
  - GalilCoupledAxes

Properties:

Device Properties		
Property name	**Property type**	**Description**
Master	Tango::DEV_STRING	letter of the master axis, from A to H This axis is the master axis

Device Properties Default Values:

Property Name	Default Values
Master	No default value

There is no Class properties.

Attributes:

Scalar Attributes			
Attribute name	**Data Type**	**R/W Type**	**Expert**
position: position of the master axis	DEV_DOUBLE	READ_WRITE	No
forwardLimitSwitch: state of the forward limit switch true if Forward LSW present	DEV_BOOLEAN	READ	No
backwardLimitSwitch: state of the backward limit switch true if Backward LSW present	DEV_BOOLEAN	READ	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
Stop	DEV_VOID	DEV_VOID
Forward	DEV_VOID	DEV_VOID
Backward	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:**

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:**

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:**

4 - Stop

- **Description:** stops the movement
- **Argin:**
  DEV_VOID :
- **Argout:**
  DEV_VOID :
- **Command allowed for:**

5 - Forward

- **Description:** initiates a jog forward continuous forward movement, stopped with Stop command
- **Argin:**
  DEV_VOID :
- **Argout:**
  DEV_VOID :
- **Command allowed for:**

6 - Backward

- **Description:** initiates a jog backward continuous backward movement, stopped with Stop command
- **Argin:**
  DEV_VOID :
- **Argout:**
  DEV_VOID :
- **Command allowed for:**




TANGO Device Server

GalilCoupledAxes User's Guide

GalilCoupledAxes Class

Revision: release_2_3_10 - Author: leclercq
Implemented in C++

Introduction:

This class is used to control coupled axes by controlling only the master axis. All the master-slave axis configuration are done by motorists in micro-code itself. Special commands if needed are accessed through MicrocodeDataViewer DS First need : a detector table with 2 motors

Class Inheritance:

- Tango::Device_3Impl
  - GalilCoupledAxes

Properties:

Device Properties		
Property name	**Property type**	**Description**
Master	Tango::DEV_STRING	letter of the master axis, from A to H This axis is the master axis

Device Properties Default Values:

Property Name	Default Values
Master	No default value

There is no Class properties.

Attributes:

Scalar Attributes			
Attribute name	**Data Type**	**R/W Type**	**Expert**
position: position of the master axis	DEV_DOUBLE	READ_WRITE	No
forwardLimitSwitch: state of the forward limit switch true if Forward LSW present	DEV_BOOLEAN	READ	No
backwardLimitSwitch: state of the backward limit switch true if Backward LSW present	DEV_BOOLEAN	READ	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
Stop	DEV_VOID	DEV_VOID
Forward	DEV_VOID	DEV_VOID
Backward	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:**

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:**

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:**

4 - Stop

- **Description:** stops the movement
- **Argin:**
  DEV_VOID :
- **Argout:**
  DEV_VOID :
- **Command allowed for:**

5 - Forward

- **Description:** initiates a jog forward continuous forward movement, stopped with Stop command
- **Argin:**
  DEV_VOID :
- **Argout:**
  DEV_VOID :
- **Command allowed for:**

6 - Backward

- **Description:** initiates a jog backward continuous backward movement, stopped with Stop command
- **Argin:**
  DEV_VOID :
- **Argout:**
  DEV_VOID :
- **Command allowed for:**

Frame Alert

This document is designed to be viewed using the frames feature. If you see this message, you are using a non-frame-capable web client.
Link to Non-frame version.




TANGO Device Server

GalilCoupledAxes
Device Commands Description
GalilCoupledAxes Class

Revision: release_2_3_10 - Author: leclercq

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:**

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:**

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:**

4 - Stop

- **Description:** stops the movement

- **Argin:**
  DEV_VOID :

- **Argout:**
  DEV_VOID :

- **Command allowed for:**

5 - Forward

- **Description:** initiates a jog forward continuous forward movement, stopped with Stop command

- **Argin:**
  DEV_VOID :

- **Argout:**
  DEV_VOID :

- **Command allowed for:**

6 - Backward

- **Description:** initiates a jog backward continuous backward movement, stopped with Stop command

- **Argin:**
  DEV_VOID :

- **Argout:**
  DEV_VOID :

- **Command allowed for:**

ESRF - Software Engineering Group