



**TANGO Device Server**

# Paragon Stepper Motor User's Guide

# Paragon Class

**Revision: Paragon-Release_2_0 - Author: vedder**
**Implemented in C++ - CVS repository: ESRF**

## Introduction:

A class to control stepper motors with the DIN Rail Stepper Drive L25i/L50i.

## Class Inheritance:

- Tango::Device_4Impl
  - Paragon

## Class Description:

# Properties:

| Device Properties | | |
|---|---|---|
| **Property name** | **Property type** | **Description** |
| **Serialline** | Tango::DEV_STRING | Serial line device to access the motor controller. |
| **Channel** | Tango::DEV_SHORT | Channel number for the motor on the controller. Number between 1 and 6. |
| **Limit_switches** | Tango::DEV_STRING | Limit switches can be ’enabled’ or ’disabled’. |
| **Acceleration** | Tango::DEV_DOUBLE | Motor Acceleration |
| **Steps_per_unit** | Tango::DEV_DOUBLE | Motors steps per unit of the position value. Position might have a unit as: mm, um, rad, mrad.... |
| **Velocity** | Tango::DEV_DOUBLE | Constant velocity of the motor |
| **Current** | Tango::DEV_LONG | Current to drive motor in |
| **Micro_stepping** | Tango::DEV_LONG | Multiplies the steps/tr of the motor by the micro_stepping value. |
| **Backlash** | Tango::DEV_DOUBLE | Backlash to be applied with every motor movement. |
| **Encoder_present** | Tango::DEV_SHORT | The motor position is read either in motor steps (=0) or in encoder steps (=1). |
| **Encoder_input** | Tango::DEV_SHORT | Refer to Paragon stepper drive user’s guide. 9=step/dir, 1=cw/ccw, 2=quad ABZ, default=2 |
| **Encoder_resolution** | Tango::DEV_LONG | Resolution of the encoder connected to the axe. |
| **Home_switch_edge** | Tango::DEV_STRING | The reference edge of the home switch to be used for the home positioning. The value can either be the rising edge (+) or falling edge (-). |
| **Home_switch_type** | Tango::DEV_SHORT | The home switch used can be normally in a open = 0 or a closed = 1 position. |
| **Home_search_direction** | Tango::DEV_STRING | Direction to search for the home switch + or -. |
| **Home_mode** | Tango::DEV_SHORT | The mode determines what happens when the specified edge of the home switch is encountered: See the stepper drive user guide for a detailed description |
| **Home_position** | Tango::DEV_DOUBLE | The position value to be applied to the motor in the home position. |
| **Posmain** | Tango::DEV_STRING | Position mainenanace can be \"enabled\" or \"disabled\". |
| **Posmain_deadband_range** | Tango::DEV_LONG | The dead band range is measured in encoder counts as a +/- band. A dead band of 10 will be 20 units wide |
| **Posmain_settle_time** | Tango::DEV_LONG | Time in milliseconds the indexer will wait after a motion, before doing a feedback. |
| **Calibrated** | Tango::DEV_SHORT | When this property is different from 0, the motor is considered as calibrated and a certain number of attributes cannot be changed anymore.( e.g. step_per_unit) The goal is to avoid undesired change when the calibratiuon process has been performed. |

| | | |
|---|---|---|
| **Position** | Tango::DEV_DOUBLE | The last position value saved to the database by the server. Must be written to the controller on start-up. |
| **Auto_numbering** | Tango::DEV_STRING | Defines if the controller is configured to accept the auto numbering. This feature is only available with RS232 controlled devices. This property is set by default to \"ON\" in the code of the server. It should be set to \"OFF\" to disable it. |
| **BrakeMask** | Tango::DEV_STRING | This is a string representing the mask to be applied to the output bits in order to activate or de-activate the brake mechanism. All the bits set to 'x' will not be changed, only the bit set to '1'. |

Device Properties Default Values:

| Property Name | Default Values |
|---|---|
| Serialline | No default value |
| Channel | No default value |
| Limit_switches | enabled |
| Acceleration | 10 |
| Steps_per_unit | 1 |
| Velocity | 1 |
| Current | 50 |
| Micro_stepping | 20 |
| Backlash | 0 |
| Encoder_present | 0 |
| Encoder_input | 2 |
| Encoder_resolution | 1 |
| Home_switch_edge | + |
| Home_switch_type | 0 |
| Home_search_direction | - |
| Home_mode | 2 |
| Home_position | 0 |
| Posmain | disabled |
| Posmain_deadband_range | 0 |
| Posmain_settle_time | 0 |
| Calibrated | 0 |
| Position | No default value |
| Auto_numbering | No default value |
| BrakeMask | XXXXXXXX |

**There is no Class properties.**

# States:

| States | |
|---|---|
| **Names** | **Descriptions** |
| **ON** | The motor powered on and is ready to move. |
| **MOVING** | The motor is moving |
| **FAULT** | The motor indicates a fault. |
| **ALARM** | The motor indicates an alarm state for example has reached a limit switch. |
| **OFF** | The power on the moror drive is switched off. |
| **DISABLE** | The motor is in slave mode and disabled for normal use |

# Attributes:

| Scalar Attributes | | | |
|---|---|---|---|
| **Attribute name** | **Data Type** | **R/W Type** | **Expert** |
| **Position**: The actual motor position. | DEV_DOUBLE | READ_WRITE | No |
| **Tweak_step**: Step size to move the motor when executing the 'Tweak' command. | DEV_DOUBLE | READ_WRITE | No |
| **Steps_per_unit** | DEV_DOUBLE | READ_WRITE | Yes |
| **Acceleration**: The acceleration of the motor. | DEV_DOUBLE | READ_WRITE | Yes |
| **Velocity**: The constant velocity of the motor. | DEV_DOUBLE | READ_WRITE | Yes |
| **Current**: The current used to drive the motor. | DEV_LONG | READ_WRITE | Yes |
| **Micro_stepping** | DEV_LONG | READ_WRITE | Yes |
| **Backlash**: Backlash to be applied to each motor movement | DEV_DOUBLE | READ_WRITE | Yes |
| **Steps** | DEV_LONG | READ | No |
| **Encoder_present**: 0 = False 1= True | DEV_SHORT | READ_WRITE | Yes |
| **Encoder_input** | DEV_SHORT | READ_WRITE | Yes |
| **Encoder_resolution** | DEV_LONG | READ_WRITE | Yes |
| **Home_switch_edge** | DEV_STRING | READ_WRITE | Yes |
| **Home_switch_type**: Pulse switch: 0=normally open; 1=normally closed. Simple switch: 3=switch from open to close. | DEV_SHORT | READ_WRITE | Yes |
| **Home_search_direction** | DEV_STRING | READ_WRITE | Yes |
| **Home_mode** | DEV_SHORT | READ_WRITE | Yes |
| **Home_position** | DEV_DOUBLE | READ_WRITE | Yes |
| **Position_error** | DEV_LONG | READ | No |
| **Posmain_deadband_range**: The dead band range is measured in encoder counts as a +/- band. A dead band of 10 will be 20 units wide! | DEV_LONG | READ_WRITE | Yes |
| **Posmain_settle_time**: Time in milliseconds the indexer will wait after a motion, before doing a feedback | DEV_LONG | READ_WRITE | Yes |
| **Limits_status**: limits_status[0] = Home (0 or 1 (or X->communication pb)) limits_status[1] = Limit - (0 or 1 (or X->communication pb)) limits_status[2] = Limit +(0 or 1 (or X->communication pb)) | DEV_STRING | READ | Yes |

# 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 |
| **Tweak** | DEV_VOID | DEV_VOID |
| **On** | DEV_VOID | DEV_VOID |
| **Off** | DEV_VOID | DEV_VOID |
| **GoHome** | DEV_VOID | DEV_VOID |
| **Abort** | DEV_VOID | DEV_VOID |
| **BrakeOn** | DEV_VOID | DEV_VOID |
| **BrakeOff** | DEV_VOID | DEV_VOID |

| Device Commands for Expert Level Only | | |
|---|---|---|
| **Command name** | **Argument In** | **Argument Out** |
| **WriteAbsolutePosition** | DEV_DOUBLE | DEV_VOID |
| **PosmainOn** | DEV_VOID | DEV_VOID |
| **PosmainOff** | 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

- Tango::ALARM
- Tango::OFF
- Tango::DISABLE

## 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
  - Tango::ALARM
  - Tango::OFF
  - Tango::DISABLE

## 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
  - Tango::ALARM
  - Tango::OFF
  - Tango::DISABLE

## 4 - Tweak

- **Description:** Tweak the motor by the value of the tweak_step attribute value.

- **Argin:**
  **DEV_VOID** :

- **Argout:**
  **DEV_VOID** :

- **Command allowed for:**
  - Tango::ON
  - Tango::ALARM

## 5 - On

- **Description:** Enable power on motor

- **Argin:**
  **DEV_VOID** :

- **Argout:**
  **DEV_VOID** :

- **Command allowed for:**
  - Tango::ON
  - Tango::ALARM
  - Tango::OFF

## 6 - Off

- **Description:** Desable power on motor

- **Argin:**
  **DEV_VOID** :

- **Argout:**
  **DEV_VOID** :

- **Command allowed for:**
  - Tango::ON
  - Tango::ALARM
  - Tango::OFF

## 7 - WriteAbsolutePosition (for expert only)

- **Description:** Define an absolute position

- **Argin:**
  **DEV_DOUBLE** :

- **Argout:**
  **DEV_VOID** :

- **Command allowed for:**
  - Tango::ON
  - Tango::ALARM
  - Tango::OFF

## 8 - GoHome

- **Description:** Move the motor to the home position given by a home switch.

- **Argin:**
  **DEV_VOID** :

- **Argout:**
  **DEV_VOID** :

- **Command allowed for:**
  - Tango::ON
  - Tango::ALARM

## 9 - PosmainOn (for expert only)

- **Description:** Switch on position maintenance feedback.

- **Argin:**
  **DEV_VOID** :

- **Argout:**
  **DEV_VOID** :

- **Command allowed for:**
  - Tango::ON
  - Tango::ALARM
  - Tango::OFF

## 10 - PosmainOff (for expert only)

- **Description:** Switch off position maintenance feedback.

- **Argin:**
  **DEV_VOID** :

- **Argout:**
  **DEV_VOID** :

- **Command allowed for:**
  - Tango::ON
  - Tango::ALARM
  - Tango::OFF

## 11 - Abort

- **Description:** Abort a movment of the motor.

- **Argin:**
  **DEV_VOID** :

- **Argout:**
  **DEV_VOID** :

- **Command allowed for:**
  - Tango::ON
  - Tango::MOVING
  - Tango::FAULT
  - Tango::ALARM
  - Tango::OFF
  - Tango::DISABLE

## 12 - BrakeOn

- **Description:** Activate the brake mechanism.

- **Argin:**
  **DEV_VOID** :

- **Argout:**
  **DEV_VOID** :

- **Command allowed for:**
  - Tango::ON
  - Tango::MOVING

- Tango::FAULT
- Tango::ALARM
- Tango::OFF
- Tango::DISABLE

## 13 - BrakeOff

- **Description:** De-activate the brake mechanism.

- **Argin:**
  **DEV_VOID** :

- **Argout:**
  **DEV_VOID** :

- **Command allowed for:**
- Tango::ON
- Tango::MOVING
- Tango::FAULT
- Tango::ALARM
- Tango::OFF
- Tango::DISABLE