


TANGO Device Server

Novelec MCCE-2 Electrometers User's Guide

MCCE2Electrometers Class

Revision: release_2_0_2 - Author: xavela
Implemented in C++

Introduction:

This class allows you to control all MCCE-2 Novelec electrometers through a Serial bus.
Supported types :
Novelec Electrometers : MCCE2

Class Inheritance:

- Tango::Device_3Impl
  - MCCE2Electrometers

Properties:

Device Properties		
Property name	**Property type**	**Description**
CommunicationLinkName	Tango::DEV_STRING	The name of the device which manage the communication.
ElectrometerType	Tango::DEV_USHORT	The elctrometer type number. -> for Type 1 Novelec model : 1

Device Properties Default Values:

Property Name	**Default Values**
CommunicationLinkName	No default value
ElectrometerType	No default value

There is no Class properties.

Attributes:

Scalar Attributes			
Attribute name	**Data Type**	**R/W Type**	**Expert**
electrometerChannel: MCCE2 electrometer have two channels to support two amplifiers. WARN : default is 0 (please set this value first).	DEV_USHORT	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
RangeUP	DEV_VOID	DEV_VOID
RangeDOWN	DEV_VOID	DEV_VOID
SetZeroVFFunction	DEV_VOID	DEV_VOID
SetOffsetZeroV1Function	DEV_VOID	DEV_VOID
SetLeakageZeroV2Function	DEV_VOID	DEV_VOID
SetTestFunction	DEV_VOID	DEV_VOID
SetMeasureFunction	DEV_VOID	DEV_VOID
MCCE2_ON	DEV_VOID	DEV_VOID
MCCE2_OFF	DEV_VOID	DEV_VOID
Local	DEV_VOID	DEV_VOID
Remote	DEV_VOID	DEV_VOID
Reset	DEV_VOID	DEV_VOID
ClearRegisters	DEV_VOID	DEV_VOID
GetRange	DEV_VOID	CONST_DEV_STRING
SetRange	DEV_STRING	DEV_VOID
GetMode	DEV_VOID	CONST_DEV_STRING
GetPolarity	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:**

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 - RangeUP

- **Description:** Up the range of the electrometer. Throw : - electrometer::ElectrometerException if the range limit is reached - Tango::DevFailed if the command cannot be performed
- **Argin:**
  DEV_VOID : no argin
- **Argout:**
  DEV_VOID : no argout
- **Command allowed for:**

5 - RangeDOWN

- **Description:** Down the range of the electrometer. Throw : - electrometer::ElectrometerException if the range is negative - Tango::DevFailed if the command cannot be performed
- **Argin:**
  DEV_VOID : no argin
- **Argout:**
  DEV_VOID : no argout
- **Command allowed for:**

6 - SetZeroVFFunction

- **Description:** Enable the Zero V/F MCCE2 mode.
- **Argin:**
  DEV_VOID : no argin
- **Argout:**
  DEV_VOID : no argout
- **Command allowed for:**

7 - SetOffsetZeroV1Function

- **Description:** Enable the Offset or Zero V1 MCCE2 mode.
- **Argin:**
  DEV_VOID : no argin
- **Argout:**
  DEV_VOID : no argout
- **Command allowed for:**

8 - SetLeakageZeroV2Function

- **Description:** Enable the Leakage or Zero V2 MCCE2 mode.
- **Argin:**
  DEV_VOID : no argin
- **Argout:**
  DEV_VOID : no argout
- **Command allowed for:**

9 - SetTestFunction

- **Description:** Enable theTest MCCE2 mode.
- **Argin:**
  DEV_VOID : no argin
- **Argout:**
  DEV_VOID : no argout
- **Command allowed for:**

10 - SetMeasureFunction

- **Description:** Enable the Measure MCCE2 mode.
- **Argin:**
  DEV_VOID : no argin
- **Argout:**
  DEV_VOID : no argout
- **Command allowed for:**

11 - MCCE2_ON

- **Description:** Enable the MEASURE function
- **Argin:**
  DEV_VOID : no argin
- **Argout:**
  DEV_VOID : no argout
- **Command allowed for:**

12 - MCCE2_OFF

- **Description:** Disable the MEASURE function
- **Argin:**
  DEV_VOID : no argin
- **Argout:**
  DEV_VOID : no argout
- **Command allowed for:**

13 - Local

- **Description:** Enable local keyboard and Key
- **Argin:**
  DEV_VOID : no argin
- **Argout:**
  DEV_VOID : no argout
- **Command allowed for:**

14 - Remote

- **Description:** Disable \"MODIFY\" and the \"ON-OFF\" keys
- **Argin:**
  DEV_VOID : no argin
- **Argout:**
  DEV_VOID : no argout
- **Command allowed for:**

15 - Reset

- **Description:** Restart the MCCE-2
- **Argin:**
  DEV_VOID : no argin
- **Argout:**
  DEV_VOID : no argout
- **Command allowed for:**

16 - ClearRegisters

- **Description:** Clear error(s)
- **Argin:**
  DEV_VOID : no argin
- **Argout:**
  DEV_VOID : no argout
- **Command allowed for:**

17 - GetRange

- **Description:** Returns the electrometer range.
- **Argin:**
  DEV_VOID : no argin
- **Argout:**
  CONST_DEV_STRING : The actual electrometer range
- **Command allowed for:**

18 - SetRange

- **Description:** Apply the specified argin range, on the electrometer, if well formatted. Else an exception is thrown.
- **Argin:**
  DEV_STRING : The range to apply on the electrometer
- **Argout:**
  DEV_VOID : no argout
- **Command allowed for:**

19 - GetMode

- **Description:** Returns the electrometer mode which can be one of the following values : MEASURE, V/F Zero, OFFSET, LEAKAGE or TEST.
- **Argin:**
  DEV_VOID : no argin
- **Argout:**
  CONST_DEV_STRING : The electrometer mode (MEASURE, LEAKAGE ...)
- **Command allowed for:**

20 - GetPolarity

- **Description:** Returns the electrometer polarity, which can be POSITIVE or NEGATIVE.
- **Argin:**
  DEV_VOID : no argin
- **Argout:**
  CONST_DEV_STRING : The electrometer polarity
- **Command allowed for:**




TANGO Device Server

Novelec MCCE-2 Electrometers User's Guide

MCCE2Electrometers Class

Revision: release_2_0_2 - Author: xavela
Implemented in C++

Introduction:

This class allows you to control all MCCE-2 Novelec electrometers through a Serial bus.
Supported types :
Novelec Electrometers : MCCE2

Class Inheritance:

- Tango::Device_3Impl
  - MCCE2Electrometers

Properties:

Device Properties		
Property name	**Property type**	**Description**
CommunicationLinkName	Tango::DEV_STRING	The name of the device which manage the communication.
ElectrometerType	Tango::DEV_USHORT	The elctrometer type number. -> for Type 1 Novelec model : 1

Device Properties Default Values:

Property Name	**Default Values**
CommunicationLinkName	No default value
ElectrometerType	No default value

There is no Class properties.

Attributes:

Scalar Attributes			
Attribute name	**Data Type**	**R/W Type**	**Expert**
electrometerChannel: MCCE2 electrometer have two channels to support two amplifiers. WARN : default is 0 (please set this value first).	DEV_USHORT	WRITE	No

Commands:

More Details on commands....