



TANGO Device Server

Generic CUB device server
User's Guide

Cubds Class

Revision: - Author: epaud
Implemented in C++ - CVS repository: ESRF

Introduction:

This class allows to access the inputs and outputs registers of the AMCC used as PCI interface on the CUB cards. It is undependant of the FPGA embedded application.

Class Description:

The device handled by this server is the entire CUB board.
It allows to access the IMBs and OMBs registers of the AMCC chip used as C-PCI interface of the CUB card.
At startup the default values of the IMBs registers are taken in the database and puts in the IMBs registers.
Also the xilinx filename to load into the embedded FPGA is taken into the database.
At first startup all the resources are created within the database with default values and needs to be tuned
afterward. Some of them can also be changed with the WriteCUBxxx() commands if required.
The server is organised with a main part which allows the communication with the clients and also a thread part which read permantly the IMB4 register of the CUB card and latches the error bits in the status register. When the status register is read the latched conditions are reset.

The IMB4 CUB's status register must be organised as follow:
- Bits 0..3: Overflow bits
- Bits 4..15: Not treated
- Bits 16..23: Internal counter used for synchronising the data
- Bits 24..31: Not treated

Tango Device Server User's Guide

Properties:

Device Properties		
Property name	**Property type**	**Description**
OMB2	Tango::DEV_LONG	Default value of OMB2. Default value is set to 0.
OMB3	Tango::DEV_LONG	Default value of OMB3. Default value is set to 0.
OMB4	Tango::DEV_LONG	Default value of OMB4.Default value is set to 0.
OMB1	Tango::DEV_LONG	Default value of OMB1. Default value is set to 0.
Devicename	Tango::DEV_STRING	CUB device name. Default value is set to '/dev/cubdev1
Virtexfilename	Tango::DEV_STRING	Name of the virtex file to be loaded on the FPGA. The default is set to 'unknow' and the server breaks if this resource is not set.
Virtexfilepath	Tango::DEV_STRING	Filepath where to find the virtex file. The default value is to '.
DebugFlag	Tango::DEV_LONG	Original value of the debug bit field used for printout.
PollingFrequency	Tango::DEV_DOUBLE	IMB4 CUB's status register polling frequency in Hz. Be carrefull to not ask a too high value which will not be possible to reach by the driver.

Device Properties Default Values:

Property Name	Default Values
OMB2	No default value
OMB3	No default value
OMB4	No default value
OMB1	No default value
Devicename	No default value
Virtexfilename	No default value
Virtexfilepath	No default value
DebugFlag	No default value
PollingFrequency	No default value

There is no Class properties.

States:

States	
Names	**Descriptions**
INIT	The server remains in this state until the FPGA is not loaded and the embedded program is not running.
RUNNING	When the FPGA has been loaded and the server has detected a known word in the CUB status register, meaning that the embeded program is running, the state is passed to RUNNING

Attributes:

Scalar Attributes			
Attribute name	**Data Type**	**R/W Type**	**Expert**
OMB2: General purpose OMB2 register. The meaning can be different for each application.	DEV_LONG	READ_WRITE	Yes
OMB3: General purpose OMB3 register. The meaning can be different for each application.	DEV_LONG	READ_WRITE	Yes
OMB4: General purpose OMB4 register. The meaning can be different for each application.	DEV_LONG	READ_WRITE	Yes
OMB1: General purpose OMB1 register. The meaning can be different for each application.	DEV_LONG	READ_WRITE	Yes

Spectrum Attributes			
Attribute name	**Data Type**	**X Data Length**	**Expert**
IMB2: General purpose IMB2 register. The meaning can be different for each application.	DEV_LONG	8192	Yes
IMB3: General purpose IMB3 register. The meaning can be different for each application.	DEV_LONG	8192	Yes
IMB1: General purpose IMB1 register. The meaning can be different for each application.	DEV_LONG	8192	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
WriteCubUShort	DEVVAR_LONGARRAY	DEV_VOID
WriteCubUChar	DEVVAR_LONGARRAY	DEV_VOID
WriteCubULong	DEVVAR_LONGARRAY	DEV_VOID
ReadCubUShort	DEVVAR_LONGARRAY	DEVVAR_USHORTARRAY
ReadCubUChar	DEVVAR_LONGARRAY	DEVVAR_USHORTARRAY
ReadCubULong	DEVVAR_LONGARRAY	DEVVAR_ULONGARRAY

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::INIT
  - Tango::RUNNING

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::INIT
  - Tango::RUNNING

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::INIT
  - Tango::RUNNING

4 - WriteCubUShort

- **Description:** This command write a word in the CUB input register (OMBs) at the specified word offset .
  The inputs parameters are the following:

- argin[0]: Register number [1..4]
- argin[1]: Offset: 1: LSW (b15..b0) , 2: MSW (b31..b16)
- argin[2]: 0: Store to resource database, This means that there is only one value
n: Number of value (not stored to the database)
- argin[3]: Value 1
- argin[4]: Value 2
::::::::::
- argin[n]: Value n

- **Argin:**
DEVVAR_LONGARRAY : The parameters has described above

- **Argout:**
DEV_VOID :

- **Command allowed for:**
  - Tango::RUNNING

5 - WriteCubUChar

- **Description:** This command write a byte in the CUB input register (OMBs) at the specified byte offset .
The inputs parameters are the following:
- argin[0]: Register number [1..4]
- argin[1]: Offset: 1: LSB (b7..b0), 2: M-LSB (b15..b8),
..... 3: M-MSB (b23..b16), 4: MSB (b31..b24)
- argin[2]: 0: Store to resource database, This means that there is only one value
n: Number of value (not stored to the database)
- argin[3]: Value 1
- argin[4]: Value 2
::::::::::
- argin[n]: Value n

- **Argin:**
DEVVAR_LONGARRAY : The parameters has described above

- **Argout:**
DEV_VOID :

- **Command allowed for:**
  - Tango::RUNNING

6 - WriteCubULong

- **Description:** This command write a long in the CUB input register (OMBs)
The inputs parameters are the following:
- argin[0]: Register number [1..4]

- argin[1]: Not used
- argin[2]: 0: Store to resource database, This means that there is only one value
n: Number of value (not stored to the database)
- argin[3]: Value 1
- argin[4]: Value 2
:::::::::
- argin[n]: Value n

- **Argin:**
DEVVAR_LONGARRAY : The parameters has described above

- **Argout:**
DEV_VOID :

- **Command allowed for:**
  - Tango::RUNNING

7 - ReadCubUShort

- **Description:** This command read one CUB output register (IMBs) at the specified word offset .
The IMB4 register is used for dialog
between the FPGA application and the server, therefore is not directly accessible,
therefore the returned value is an image of a value previously read by the server.
The inputs parameters are the following:
- argin[0]: Register number [1..4]
- argin[1]: Offset: 1: LSW (b15..b0) , 2: MSW (b31..b16)
- argin[2]: Number of loops the register is read

- **Argin:**
DEVVAR_LONGARRAY : The parameters has described above

- **Argout:**
DEVVAR_USHORTARRAY : Array of unsigned short values

- **Command allowed for:**
  - Tango::RUNNING

8 - ReadCubUChar

- **Description:** This command read CUB output register (IMBs) at the specified byte offset . <\br>
The IMB4 register is used for dialog
between the FPGA application and the server, therefore is not directly accessible,
therefore the returned value is an image of a value previously read by the server.
The inputs parameters are the following:
- argin[0]: Register number [1..4]
- argin[1]: Offset: 1: LSB (b7..b0), 2: M-LSB (b15..b8),
..... 3: M-MSB (b23..b16), 4: MSB (b31..b24)

- argin[2]: Number of loops the register is read
On return bits 8 to 15 are forced to 0.

- **Argin:**
DEVVAR_LONGARRAY : The parameters has described above

- **Argout:**
DEVVAR_USHORTARRAY : Array of unsigned char values

- **Command allowed for:**
  - Tango::RUNNING

9 - ReadCubULong

- **Description:** This command read one CUB output register (IMBs).
The IMB4 register is used for dialog
between the FPGA application and the server, therefore is not directly accessible,
therefore the returned value is an image of a value previously read by the server.
The inputs parameters are the following:
- argin[0]: Register number [1..4]
- argin[1]: Not used
- argin[2]: Number of loops the register is read

- **Argin:**
DEVVAR_LONGARRAY : The parameters has described above

- **Argout:**
DEVVAR_ULONGARRAY : Array of unsigned long values

- **Command allowed for:**
  - Tango::RUNNING