TANGO
Device
Server

Single Channel ICV712 Digital Analogic converter User’s Guide

SDac712 Class

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

Introduction:

This device server will control any channel of the ADAS ICV712 digital to analog converter board.

Class Inheritance:

- Tango::Device_4Impl
  - SDac712

Class Description:

Properties:

Device Properties		
Property name	**Property type**	**Description**
Channel	Tango::DEV_SHORT	Number of the channel (starting from 0, max value corresponds to the max number of channels supported by the card minus one. This is a mandatory property. The server will not continue the startup if not set.
Card	Tango::DEV_STRING	Name of the icv712 board device (Dac712 class). Through this property, the link is made between the card (device descriptor) and the channel. This is a mandatory property, the server will not continue the startup if not set.
Minvalue	Tango::DEV_DOUBLE	Minimum value corresponding to a 0 value read on the channel.
Maxvalue	Tango::DEV_DOUBLE	Maximum value corresponding to 4095 read on the channel.

Device Properties Default Values:

Property Name	Default Values
Channel	No default value
Card	No default value
Minvalue	No default value
Maxvalue	No default value

There is no Class properties.

States:

States	
Names	**Descriptions**
ON	The device is On and working. All commands are allowed.
FAULT	There is a problem accessing the card or the device driver. The state remains in this state until a successfull access has been made. All commands are allowed, but if the state remains in fault, their effect is unpredictable.

Attributes:

Scalar Attributes			
Attribute name	**Data Type**	**R/W Type**	**Expert**
Value	DEV_DOUBLE	READ_WRITE	No
MinValue	DEV_DOUBLE	READ_WRITE	Yes
MaxValue	DEV_DOUBLE	READ_WRITE	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
Reset	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::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::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::FAULT

4 - Reset

- **Description:** reset the board.

- **Argin:**
  DEV_VOID :

- **Argout:**
  DEV_VOID :

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