TANGO
Device
Server

PSGPowerSupplyCrate
User's Guide

PSGPowerSupplyCrate Class

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

Introduction:

This class is able to control the ESRF Power Supply for administrate Crate It uses other classes: * PowerSupplyChannel class * SRCorrectors class * MchCorrector class

Class Identification:

- **Contact :** at esrf.fr - yohannlavigne
- **Class Family :** PowerSupply (Specific Specific)
- **Platform :** Unix Like
- **Bus :** Not Applicable
- **Manufacturer :** ESRF

Class Inheritance:

- Tango::Device_4Impl
    - PSGPowerSupplyCrate

Properties:

Device Properties		
Property name	**Property type**	**Description**
FBusChannel	Tango::DEV_STRING	FBus Channel
Node	Tango::DEV_STRING	Node number
Crate	Tango::DEV_STRING	Crate A or B.

Device Properties Default Values:

Property Name	Default Values
FBusChannel	/dev/fbus_00
Node	0
Crate	No default value

There is no Class properties.

Attributes:

Scalar Attributes			
Attribute name	**Data Type**	**R/W Type**	**Expert**
TRI: Three-phase enable or disable.	DEV_BOOLEAN	READ	No
Crate: Crate ON or OFF.	DEV_BOOLEAN	READ	No
Waterflow: Waterflow enable or disable.	DEV_BOOLEAN	READ	No
PresencePlus18V: 18V enable or disable.	DEV_BOOLEAN	READ	No
PresenceMinus18V: -18V enable or disable.	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
Reset	DEV_VOID	DEV_VOID
CrateON	DEV_VOID	DEV_VOID
CrateOFF	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 - Reset

- **Description:** Reset the powersupply to a well known state.

- **Argin:**
  DEV_VOID :

- **Argout:**
  DEV_VOID :

- **Command allowed for:**

5 - CrateON

- **Description:** Power ON Crate

- **Argin:**
  DEV_VOID : Power ON Crate

- **Argout:**
  DEV_VOID :

- **Command allowed for:**

6 - CrateOFF

- **Description:** Power OFF Crate

- **Argin:**
  DEV_VOID : Power OFF Crate

- **Argout:**
  DEV_VOID :

- **Command allowed for:**