



TANGO
Device
Server

ProfibusServer User's Guide

ProfibusServer Class

**Revision: release_1_0_0 - Author: coquet
Implemented in C++**

Introduction:

handles Hilscher Profibus cards, CIF80PB and CIF 60PB this server can work with up to 4 cards
this server is also called ProfibusMultiServer

Class Inheritance:

- Tango::Device_3Impl
 - ProfibusServer

Properties:

Device Properties		
Property name	Property type	Description
Board0Address	Tango::DEV_LONG	Board 0 DP address default 0 master board is assigned Pprofibus DP address 0
Board1Address	Tango::DEV_LONG	Board 1 DP address default -1 for board 1 not present
Board2Address	Tango::DEV_LONG	Board 2 DP address default -1 for board 1 not present
Board3Address	Tango::DEV_LONG	Board 3 DP address default -1 for board 1 not present
Board0ConfigurationFile	Tango::DEV_STRING	configuration file of the master board # 0 default : tango/profibus/board0
Board1ConfigurationFile	Tango::DEV_STRING	configuration file of the master board # 1 default :
Board2ConfigurationFile	Tango::DEV_STRING	configuration file of the master board #2 default :
Board3ConfigurationFile	Tango::DEV_STRING	configuration file of the master board #3 default :

Device Properties Default Values:

Property Name	Default Values
Board0Address	No default value
Board1Address	No default value
Board2Address	No default value
Board3Address	No default value
Board0ConfigurationFile	No default value
Board1ConfigurationFile	No default value
Board2ConfigurationFile	No default value
Board3ConfigurationFile	No default value

There is no Class properties.

States:

States

Names	Descriptions
ON	*
OFF	*
FAULT	*
ALARM	

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
RegisterSlave	DEVVAR_LONGSTRINGARRAY	DEV_LONG
GetSlaveParameters	DEV_LONG	DEVVAR_STRINGARRAY
ReadSlaveInputs	DEV_LONG	DEVVAR_CHARARRAY
ReadSlaveOutputs	DEV_LONG	DEVVAR_CHARARRAY
WriteSlaveOutputs	DEVVAR_LONGARRAY	DEV_LONG
Sync	DEVVAR_LONGARRAY	DEV_VOID
UnSyncAll	DEV_VOID	DEV_VOID
ResetBoard	DEV_LONG	DEV_VOID
GetSlaveDiag	DEVVAR_LONGARRAY	DEVVAR_STRINGARRAY
GetFieldbusDiag	DEV_LONG	DEVVAR_STRINGARRAY
LoadProfibusConfiguration	DEV_LONG	DEV_BOOLEAN
GetBoardInfo	DEV_VOID	DEVVAR_STRINGARRAY
UnSync	DEVVAR_LONGARRAY	DEV_VOID
GetAllSlavesParameters	DEV_VOID	DEVVAR_STRINGARRAY

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

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

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

4 - RegisterSlave

- **Description:** slave is registered in a local database, and slave commands on ProfibusServer will use the ident number returned by this method. Used to simplify commands on the server and optimisation purposes string : slave device name lvalue 0 : board number lvalue 1 : slave DP address lvalue 2 : sync : freeze group lvalue 3 : input address offset lvalue 4 : input address length lvalue 5 : output address offset lvalue 6 : output address length

- **Argin:**

DEVVAR_LONGSTRINGARRAY : parameters of the slave see method description

- **Argout:**

DEV_LONG : number of the slave for commands

- **Command allowed for:**

- Tango::ON
- Tango::OFF
- Tango::FAULT
- Tango::ALARM

5 - GetSlaveParameters

- **Description:** returns a string array with the ident data of the slave or NO RECORD FOUND for slave id < 0 if no data found for ident

- **Argin:**

DEV_LONG : ident of the slave

- **Argout:**

DEVVAR_STRINGARRAY : string array with ident data of the slave

- **Command allowed for:**

- Tango::ON
- Tango::OFF
- Tango::FAULT
- Tango::ALARM

6 - ReadSlaveInputs

- **Description:** returns a char array containing the total data of the slave (as described in the registration data) parameterized as input offset, input length
- **Argin:**
DEV_LONG : ident of the slave
- **Argout:**
DEVVAR_CHARARRAY : char array containing slave inputs
- **Command allowed for:**
 - Tango::ON
 - Tango::ALARM

7 - ReadSlaveOutputs

- **Description:** returns the outputs of the slave as previously written in the dual port ram reads the outputs back from the dual port ram, not from the slave
- **Argin:**
DEV_LONG : ident of the slave
- **Argout:**
DEVVAR_CHARARRAY : array of char contains the outputs of the slave
- **Command allowed for:**
 - Tango::ON
 - Tango::ALARM

8 - WriteSlaveOutputs

- **Description:** writes the data given in argin[2]... argin[n] in unsigned char format. argin[0] is the slave ident argin[1] is the offset added to base output offset the length is given by the number of data 2...n : n - 2
- **Argin:**
DEVVAR_LONGARRAY : argin 0 : slave ident
- **Argout:**
DEV_LONG : argin 1 : offset on base output address
- **Command allowed for:**
 - Tango::ON
 - Tango::ALARM

9 - Sync

- **Description:** freezes the output of the slaves belonging to (board, group) group = 0 : to all of the groups for slaves Sync compatible only.
- **Argin:**
DEVVAR_LONGARRAY : argin[0] : board number
- **Argout:**
DEV_VOID : argin[1] : group number
- **Command allowed for:**
 - Tango::ON
 - Tango::ALARM

10 - UnSyncAll

- **Description:** unsync all slaves previously syncronized, the values previously writed during Sync turn on their outputs.
- **Argin:**
DEV_VOID :
- **Argout:**
DEV_VOID :
- **Command allowed for:**
 - Tango::ON
 - Tango::ALARM

11 - ResetBoard

- **Description:** resets the board number COLDSTART way. must be called asynchronously since reset can last 2 to 10 seconds
- **Argin:**
DEV_LONG : board number
- **Argout:**
DEV_VOID :
- **Command allowed for:**
 - Tango::ON
 - Tango::FAULT
 - Tango::ALARM

12 - GetSlaveDiag

- **Description:** returns a DevVarStringArray with the description of the slave standard diagnostic
- **Argin:**
DEVVAR_LONGARRAY : [0] card number, [1] DP Address
- **Argout:**
DEVVAR_STRINGARRAY : standard diagnostic of the slave
- **Command allowed for:**
 - Tango::ON
 - Tango::ALARM

13 - GetFieldbusDiag

- **Description:** get standard Profibus DP diagnostics
- **Argin:**
DEV_LONG : master card number
- **Argout:**
DEVVAR_STRINGARRAY :
- **Command allowed for:**
 - Tango::ON
 - Tango::FAULT
 - Tango::ALARM

14 - LoadProfibusConfiguration

- **Description:** loads the Profibus DP configuration file given in property Board(x)ConfigurationFile
- **Argin:**
DEV_LONG : board number(0 to 3)
- **Argout:**
DEV_BOOLEAN :
- **Command allowed for:**
 - Tango::ON
 - Tango::FAULT
 - Tango::ALARM

15 - GetBoardInfo

- **Description:** retourne les infos cartes maitres disponibles
- **Argin:**
DEV_VOID :
- **Argout:**
DEVVAR_STRINGARRAY : description des cartes disponibles
- **Command allowed for:**
 - Tango::ON
 - Tango::ALARM

16 - UnSync

- **Description:**
- **Argin:**
DEVVAR_LONGARRAY : [0] : the board number, [1] : the group
- **Argout:**
DEV_VOID :
- **Command allowed for:**
 - Tango::ON
 - Tango::OFF
 - Tango::FAULT
 - Tango::ALARM

17 - GetAllSlavesParameters

- **Description:**
- **Argin:**
DEV_VOID :
- **Argout:**
DEVVAR_STRINGARRAY : registered informations of clients
- **Command allowed for:**
 - Tango::ON
 - Tango::OFF
 - Tango::FAULT
 - Tango::ALARM

ESRF - Software Engineering Group



TANGO
Device
Server

ProfibusServer User's Guide

ProfibusServer Class

Revision: release_1_0_0 - Author: coquet
Implemented in C++

Introduction:

handles Hilscher Profibus cards, CIF80PB and CIF 60PB this server can work with up to 4 cards
this server is also called ProfibusMultiServer

Class Inheritance:

- Tango::Device_3Impl
 - ProfibusServer

Properties:

Device Properties		
Property name	Property type	Description
Board0Address	Tango::DEV_LONG	Board 0 DP address default 0 master board is assigned Pprofibus DP address 0
Board1Address	Tango::DEV_LONG	Board 1 DP address default -1 for board 1 not present
Board2Address	Tango::DEV_LONG	Board 2 DP address default -1 for board 1 not present
Board3Address	Tango::DEV_LONG	Board 3 DP address default -1 for board 1 not present
Board0ConfigurationFile	Tango::DEV_STRING	configuration file of the master board # 0 default : tango/profibus/board0
Board1ConfigurationFile	Tango::DEV_STRING	configuration file of the master board # 1 default :
Board2ConfigurationFile	Tango::DEV_STRING	configuration file of the master board #2 default :
Board3ConfigurationFile	Tango::DEV_STRING	configuration file of the master board #3 default :

Device Properties Default Values:

Property Name	Default Values
Board0Address	No default value
Board1Address	No default value
Board2Address	No default value
Board3Address	No default value
Board0ConfigurationFile	No default value
Board1ConfigurationFile	No default value
Board2ConfigurationFile	No default value
Board3ConfigurationFile	No default value

There is no Class properties.

States:

States

Names	Descriptions
ON	*
OFF	*
FAULT	*
ALARM	

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
RegisterSlave	DEVVAR_LONGSTRINGARRAY	DEV_LONG
GetSlaveParameters	DEV_LONG	DEVVAR_STRINGARRAY
ReadSlaveInputs	DEV_LONG	DEVVAR_CHARARRAY
ReadSlaveOutputs	DEV_LONG	DEVVAR_CHARARRAY
WriteSlaveOutputs	DEVVAR_LONGARRAY	DEV_LONG
Sync	DEVVAR_LONGARRAY	DEV_VOID
UnSyncAll	DEV_VOID	DEV_VOID
ResetBoard	DEV_LONG	DEV_VOID
GetSlaveDiag	DEVVAR_LONGARRAY	DEVVAR_STRINGARRAY
GetFieldbusDiag	DEV_LONG	DEVVAR_STRINGARRAY
LoadProfibusConfiguration	DEV_LONG	DEV_BOOLEAN
GetBoardInfo	DEV_VOID	DEVVAR_STRINGARRAY
UnSync	DEVVAR_LONGARRAY	DEV_VOID
GetAllSlavesParameters	DEV_VOID	DEVVAR_STRINGARRAY

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

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

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:**