



TANGO
Device
Server

Digital Delay Generator T560 User's Guide

DigitalDelayGenerator Class

Revision: release_1_1_0 - Author: elattaoui
Implemented in C++ - CVS repository: tango-ds

Introduction:

This device controls a T560 Digital Delay Generator : - four TTL-level pulse outputs, each programmable for delay and width up to 10 seconds each, with 10-picosecond resolution. - Programmable-level trigger input with divide/burst features and trigger enable GATE input. Controlled via a RS232 serial interface or a 10/100-mbps Ethernet interface. NOTE : - RS232 config : 38400, 8, N, 1 - ETHERNET port : 2000

Class Identification:

- **Contact** : at synchrotron-soleil.fr - elattaoui
- **Class Family** : Instrumentation
- **Platform** : All Platforms
- **Bus** : Ethernet
- **Manufacturer** : Highland Technology
- **Reference** : T560

Class Inheritance:

- Tango::Device_4Impl
 - DigitalDelayGenerator

Class Description:

This device controls a T560 Digital Delay Generator : - four TTL-level pulse outputs, each programmable for delay and width up to 10 seconds each, with 10-picosecond resolution. - Programmable-level trigger input with divide/burst features and trigger enable GATE input. Controlled via a RS232 serial interface or a 10/100-mbps Ethernet interface.

Properties:

Device Properties		
Property name	Property type	Description
CommunicationProxy	Tango::DEV_STRING	Name of the device which controls the communication link. This module support both RS232 and Ethernet interfaces.

Device Properties Default Values:

Property Name	Default Values
CommunicationProxy	noCommunication/ProxyName/Defined

There is no Class properties.

States:

States	
Names	Descriptions
FAULT	No communication with the T560 module.
ALARM	The T560 module reports error(s).
ON	The T560 module is OK.

Attributes:

Scalar Attributes			
Attribute name	Data Type	R/W Type	Expert
shotsCounter: Query shot counter	DEV_ULONG	READ	No
pulseDelayChannelA: Channel A pulse delay (unit : nanosecond).	DEV_DOUBLE	READ_WRITE	No
pulseWidthChannelA: Channel A pulse width	DEV_DOUBLE	READ_WRITE	No
activateChannelA: TRUE : enable channel A output FALSE: disable channel A output	DEV_BOOLEAN	READ_WRITE	Yes
pulseDelayChannelB: Channel B pulse delay (unit : nanosecond).	DEV_DOUBLE	READ_WRITE	No
pulseWidthChannelB: Channel B pulse width	DEV_DOUBLE	READ_WRITE	No
activateChannelB: TRUE : enable channel B output FALSE: disable channel B output	DEV_BOOLEAN	READ_WRITE	Yes
pulseDelayChannelC: Channel C pulse delay (unit : nanosecond).	DEV_DOUBLE	READ_WRITE	No
pulseWidthChannelC: Channel C pulse width	DEV_DOUBLE	READ_WRITE	No
activateChannelC: TRUE : enable channel C output FALSE: disable channel C output	DEV_BOOLEAN	READ_WRITE	Yes
pulseDelayChannelD: Channel D pulse delay (unit : nanosecond).	DEV_DOUBLE	READ_WRITE	No
pulseWidthChannelD: Channel D pulse width	DEV_DOUBLE	READ_WRITE	No
activateChannelD: TRUE : enable channel D output FALSE: disable channel D output	DEV_BOOLEAN	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
HardwareReset	DEV_VOID	DEV_VOID
ClearErrors	DEV_VOID	DEV_VOID
SaveModuleConfiguration	DEV_VOID	DEV_VOID

Device Commands for Expert Level Only		
Command name	Argument In	Argument Out
ResetShots	DEV_VOID	DEV_VOID
RecallModuleConfiguration	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::FAULT
 - Tango::ALARM
 - Tango::ON

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

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

4 - HardwareReset

- **Description:** Performs a hardware reset/restart of the T560 (equivalent to a power off/on cycle. NOTE : the reset takes about 4 seconds, after which the T560 will respond with the following string "Highland Thechnology T560 DDG " The last-saved setup will be installed.

- **Argin:**
DEV_VOID : no argin

- **Argout:**
DEV_VOID : no argout

- **Command allowed for:**

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

5 - ClearErrors

- **Description:** Clear error flag.
- **Argin:**
DEV_VOID : no argin
- **Argout:**
DEV_VOID : no argout
- **Command allowed for:**
 - Tango::FAULT
 - Tango::ALARM
 - Tango::ON

6 - ResetShots (for expert only)

- **Description:** Reset shot counter.
- **Argin:**
DEV_VOID : no argin
- **Argout:**
DEV_VOID : no argout
- **Command allowed for:**
 - Tango::FAULT
 - Tango::ALARM
 - Tango::ON

7 - SaveModuleConfiguration

- **Description:** Save the T560 module configuration in its intrenal flash memory. This configuartion is applied after a reset.
- **Argin:**
DEV_VOID : no argin
- **Argout:**
DEV_VOID : no argout
- **Command allowed for:**
 - Tango::FAULT
 - Tango::ALARM
 - Tango::ON

8 - RecallModuleConfiguration (for expert only)

- **Description:** Reapply the previous module configuration. No need to do a reset.
- **Argin:**
DEV_VOID : no argin
- **Argout:**
DEV_VOID : no argout
- **Command allowed for:**
 - Tango::FAULT
 - Tango::ALARM
 - Tango::ON

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