Delay generator DG645 device server.
User's Guide

StanfordDG645 Class

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

Introduction:

This device server is intended to control a SRS DG645 delay generator.

Class Identification:

- **Contact :** at esrf.fr - peru
- **Class Family :** Instrumentation
- **Platform :** Unix Like
- **Bus :** Ethernet
- **Manufacturer :** Stanford Research Systems
- **Reference :** DG645

Class Inheritance:

- Tango::Device_4Impl
  - StanfordDG645

Properties:

Device Properties		
Property name	**Property type**	**Description**
IpAddress	Tango::DEV_STRING	The IP address of the device.

Device Properties Default Values:

Property Name	Default Values
IpAddress	No default value

There is no Class properties.

Attributes:

Scalar Attributes			
Attribute name	**Data Type**	**R/W Type**	**Expert**
RisingDelay: Select the delay to be applied between the reference signal and the rising edge.	DEV_DOUBLE	READ_WRITE	No
RisingReference: Select the reference signal for the rising edge of the channel. Choice to be made in 't0', 'a', 'b', 'c', 'd', 'e', 'f', 'g', 'h'.	DEV_STRING	READ_WRITE	No
FallingDelay: Select the delay to be applied between the reference signal and the falling edge.	DEV_DOUBLE	READ_WRITE	No
FallingReference: Select the reference signal for the falling edge of the channel. Choice to be made in 't0', 'a', 'b', 'c', 'd', 'e', 'f', 'g', 'h'.	DEV_STRING	READ_WRITE	No
CurrentOutput: Select the output which parameters will be displayed. Choice to be made in 'ab', 'cd', 'ef', 'gh'.	DEV_STRING	READ_WRITE	No
RisingDelayStepSize: Define the delay step value for the rising edge. Use the commands 'RisingDelayStepDown' or 'RisingDelayStepUp' to apply the step.	DEV_DOUBLE	READ_WRITE	No
FallingDelayStepSize: Define the delay step value for the falling edge. Use the commands 'FallingDelayStepDown' or 'FallingDelayStepUp' to apply the step.	DEV_DOUBLE	READ_WRITE	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
ToggleDebugMode	DEV_VOID	DEV_VOID
DisableAdvancedTrigMode	DEV_VOID	DEV_VOID
TrigOnExternalRisingEdges	DEV_VOID	DEV_VOID
SetBurstModeOff	DEV_VOID	DEV_VOID
SetTriggerLevel	DEV_DOUBLE	DEV_VOID
SetCurrentOutput	DEV_STRING	DEV_VOID
RisingDelayStepDown	DEV_VOID	DEV_VOID
RisingDelayStepUp	DEV_VOID	DEV_VOID
FallingDelayStepUp	DEV_VOID	DEV_VOID
FallingDelayStepDown	DEV_VOID	DEV_VOID
SaveSettings	DEV_SHORT	DEV_VOID
RecallSettings	DEV_SHORT	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 - ToggleDebugMode

- **Description:**

- **Argin:**
  DEV_VOID :

- **Argout:**
  DEV_VOID :

- **Command allowed for:**

5 - DisableAdvancedTrigMode

- **Description:**

- **Argin:**
  DEV_VOID :

- **Argout:**
  DEV_VOID :

- **Command allowed for:**

6 - TrigOnExternalRisingEdges

- **Description:**

- **Argin:**
  DEV_VOID :

- **Argout:**
  DEV_VOID :

- **Command allowed for:**

7 - SetBurstModeOff

- **Description:**

- **Argin:**
  DEV_VOID :

- **Argout:**
  DEV_VOID :

- **Command allowed for:**

8 - SetTriggerLevel

- **Description:** The required parameter is the voltage value of the level.

- **Argin:**
  DEV_DOUBLE :

- **Argout:**
  DEV_VOID :

- **Command allowed for:**

9 - SetCurrentOutput

- **Description:** Determine which is the output whose parameters are presently available. To be chosen in (ab, cd, ef, gh).

- **Argin:**
  DEV_STRING :

- **Argout:**
  DEV_VOID :

- **Command allowed for:**

10 - RisingDelayStepDown

- **Description:**

- **Argin:**
  DEV_VOID :

- **Argout:**
  DEV_VOID :

- **Command allowed for:**

11 - RisingDelayStepUp

- **Description:**

- **Argin:**
  DEV_VOID :

- **Argout:**
  DEV_VOID :

- **Command allowed for:**

12 - FallingDelayStepUp

- **Description:**

- **Argin:**
  DEV_VOID :

- **Argout:**
  DEV_VOID :

- **Command allowed for:**

13 - FallingDelayStepDown

- **Description:**

- **Argin:**
  DEV_VOID :

- **Argout:**
  DEV_VOID :

- **Command allowed for:**

14 - SaveSettings

- **Description:** Save the DG645 settings to memory bank (from 1 to 9).

- **Argin:**
  DEV_SHORT :

- **Argout:**
  DEV_VOID :

- **Command allowed for:**

15 - RecallSettings

- **Description:** Recall settings from memory bank (1 to 9)

- **Argin:**
  DEV_SHORT :

- **Argout:**
  DEV_VOID :

- **Command allowed for:**