



TANGO
Device
Server

Linac Radio Frequency User's Guide

LinacRF Class

Revision: LinacRF-Release_1_4 - Author: vedder
Implemented in C++ - CVS repository: ESRF

Introduction:

This class will manage the Linac RadioFrequency. The three structures of the linac (buncher - section 1 - section 2) consist of coupled wave guide cavities, through which a high frequency electromagnetic wave passes at the same speed than the electrons. The electrons are adjusted to the peak of the RF wave, they are then subjected to the maximum electrical field. The phase of the section1 is the reference; phases of the RF in buncher and section 2 are matched to get the electrons adjusted to the peak of the RF wave.

Class Inheritance:

- Tango::Device_4Impl
 - LinacRF

Class Description:

Properties:

Device Properties		
Property name	Property type	Description
Bunchphase	Tango::DEV_STRING	Name of the buncher phase device.
Daresbury	Tango::DEV_STRING	Name of the daresbury device.
Delay_name	Tango::DEV_STRING	Name of the device used to get and set the Delay attribute.
Interlocks_list	Array of double	list of interlocks.
Mastersource	Tango::DEV_STRING	Name of the Tango Mastersource device.
Pbunchatt_adc	Tango::DEV_STRING	Name of the prebuncher attenuation ADC device.
Pbunchatt_dac	Tango::DEV_STRING	Name of the prebuncher attenuation DAC device.
Pbunchphase	Tango::DEV_STRING	Name of the pbuncher phase device.
Secphase_dac	Tango::DEV_STRING	Name of the section 2 phase DAC device.
Stateonnum	Tango::DEV_DOUBLE	number of the interlock relay used to determine whether the RF is on or not
Statepermnum	Tango::DEV_DOUBLE	Daresbury relay number which indicates if the device is allowed to be switched On.
Bunchphase_adc	Tango::DEV_STRING	Name of the Buncher Phase Adc device. This property is not used if the phase is controlled by a Paragon. It is used only with the vpap controller.
Offrelay	Tango::DEV_STRING	Name of the relay used to switch off RF.
Onrelay	Tango::DEV_STRING	Name of the relay used to switch on RF.
Pulsetime	Tango::DEV_LONG	Duration in milliseconds of the pulse to do on the pulsed relays.
Modulator	Tango::DEV_STRING	Name of the Modulator device used in the reset command. Reset command will call the Reset command of the modulator.

Device Properties Default Values:

Property Name	Default Values
Bunchphase	No default value
Daresbury	No default value
Delay_name	No default value
Interlocks_list	No default value
Mastersource	No default value
Pbunchatt_adc	No default value
Pbunchatt_dac	No default value
Pbunchphase	No default value
Secphase_dac	No default value
Stateonnum	No default value
Statepermnum	No default value
Bunchphase_adc	No default value
Offrelay	No default value
Onrelay	No default value
Pulsetime	No default value
Modulator	elin/mod1/run

There is no Class properties.

States:

States	
Names	Descriptions
ON	*
OFF	*
DISABLE	*
FAULT	*
UNKNOWN	

Attributes:

Scalar Attributes			
Attribute name	Data Type	R/W Type	Expert
Frequency: Frequency	DEV_DOUBLE	READ_WRITE	No
Delay: Delay	DEV_DOUBLE	READ_WRITE	No
PBunchPhase: PreBuncher Phase	DEV_DOUBLE	READ_WRITE	No
PBunchAttenu: PreBuncher Attenuation	DEV_DOUBLE	READ_WRITE	No
BunchPhase: Buncher Phase	DEV_DOUBLE	READ_WRITE	No
Sect2Phase: Section 2 Phase.	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
On	DEV_VOID	DEV_VOID
Off	DEV_VOID	DEV_VOID
Reset	DEV_VOID	DEV_VOID
TimingOn	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::OFF
- Tango::DISABLE
- Tango::FAULT
- Tango::UNKNOWN

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::DISABLE
- Tango::FAULT
- Tango::UNKNOWN

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::DISABLE
- Tango::FAULT

- Tango::UNKNOWN

4 - On

- **Description:**
- **Argin:**
DEV_VOID :
- **Argout:**
DEV_VOID :
- **Command allowed for:**
 - Tango::ON
 - Tango::OFF
 - Tango::DISABLE
 - Tango::FAULT
 - Tango::UNKNOWN

5 - Off

- **Description:**
- **Argin:**
DEV_VOID :
- **Argout:**
DEV_VOID :
- **Command allowed for:**
 - Tango::ON
 - Tango::OFF
 - Tango::DISABLE
 - Tango::FAULT
 - Tango::UNKNOWN

6 - Reset

- **Description:**
- **Argin:**
DEV_VOID :
- **Argout:**
DEV_VOID :

- **Command allowed for:**

- Tango::ON
- Tango::OFF
- Tango::DISABLE
- Tango::FAULT
- Tango::UNKNOWN

7 - TimingOn

- **Description:** This command will switch on the timing. (will issue a DevOn on the shorttiming device).

- **Argin:**

DEV_VOID :

- **Argout:**

DEV_VOID :

- **Command allowed for:**

- Tango::ON
- Tango::OFF
- Tango::DISABLE
- Tango::FAULT
- Tango::UNKNOWN

<p>TANGO is an open source project hosted by :</p> 	<p>Core and Tools : CVS repository on tango-cs project Device Servers : CVS repository on tango-ds project</p>
--	--