



**TANGO**  
Device  
Server

# **SY Tune Monitor User's Guide**

## **SyTune Class**

**Revision: SyTune-Release\_2\_1 - Author: meyer**  
**Implemented in C++ - CVS repository: ESRF**

### **Introduction:**

This class implements the tune measurement for the booster synchrotron. A number of tune values should be calculated for an acceleration cycle of the booster. A shaker is used to excite the beam and the data is taken from the turn by turn beam position measurement of Libera devices. Two devices are used. One for the measurement in the horizontal plane and another one for the vertical plane.

### **Class Inheritance:**

- Tango::Device\_4Impl
  - SyTune

## Class Description:

This class implements the tune measurement for the booster synchrotron. A number of tune values should be calculated for an acceleration cycle of the booster. A shaker is used to excite the beam and a CPCI412 data acquisition card is used.

## Properties:

<b>Device Properties</b>		
<b>Property name</b>	<b>Property type</b>	<b>Description</b>
<b>TunePoints</b>	Tango::DEV_LONG	The number of tunes to be calculated for an acceleration cycle.
<b>FftPoints</b>	Tango::DEV_LONG	The number of data values to calculate the FFT from.
<b>BorderExclusion</b>	Tango::DEV_LONG	The number of values to exclude from the border of the spectrum before searching the maximum.
<b>PeakPoints</b>	Tango::DEV_LONG	The number of points around the maximum to be used to do the fitting.
<b>FittingAlgorithm</b>	Tango::DEV_STRING	The algorithm to be used to fit the maximum peak froung in the spectrum. Parabol = linear fitting algorithm. A parabula is fitted to the peak. Gauss = non-linear algorithm. A gaussian is fitted to the peak.
<b>ShakerDevice</b>	Tango::DEV_STRING	Name of the shaker device to be used.
<b>HorizontalLiberaDevice</b>	Tango::DEV_STRING	The name of the Libera BPM device to be used for data acquisition in the horizontal plane.
<b>VerticalLiberaDevice</b>	Tango::DEV_STRING	The name of the Libera BPM device to be used for data acquisition in the vertical plane.
<b>RfAmpliDevice</b>	Tango::DEV_STRING	Name of the RF power amplifier to be used.
<b>TimerDevice</b>	Tango::DEV_STRING	Nane of the delay line device to be used

Device Properties Default Values:

Property Name	Default Values
TunePoints	50
FftPoints	1024
BorderExclusion	20
PeakPoints	3
FittingAlgorithm	gauss
ShakerDevice	No default value
HorizontalLiberaDevice	No default value
VerticalLiberaDevice	No default value
RfAmpliDevice	No default value
TimerDevice	No default value

**There is no Class properties.**

## States:

<b>States</b>	
<b>Names</b>	<b>Descriptions</b>
<b>ON</b>	The tune measurement is running. The shaker is switched on and the acquisition on the Libera BPM is started.
<b>OFF</b>	The tune measurement is stopped. The shaker is switched off an the acquisition on the Libera BPM is stopped.
<b>STANDBY</b>	The tune measurement is running to take reference (background) spectra. The shaker is swiched off and the acquisition on the Libera BPM is started.
<b>FAULT</b>	A fault was detected on the shaker, the Libera BPM or in the calculations.
<b>UNKNOWN</b>	The connection with the shaker or the Libera BPM was lost.

## Attributes:

## Scalar Attributes

Attribute name	Data Type	R/W Type	Expert
<b>ShakerOn:</b> Indicates whether the shaker is switched on or off	DEV_BOOLEAN	READ_WRITE	No
<b>ShakerMode:</b> The mode of the shaker. Can be pulsed mode or white noise mode. true = white noise, false = pulsed	DEV_BOOLEAN	READ_WRITE	No
<b>RfGain:</b> The RF gain level of the power amplifier.	DEV_SHORT	READ_WRITE	No
<b>MeasurementPlane:</b> Indicates the measurement plane used. Possible values are horizontal or vertical. True = horizontal, false = vertical	DEV_BOOLEAN	READ_WRITE	No
<b>Average:</b> The coefficient used for the linear averaging of the spectra calculated. 1 = no averaging, 100 = averaging with a coefficient of 0.01.	DEV_LONG	READ_WRITE	No
<b>Complement:</b> Switches the tunes values to a value of 1-tune when true.	DEV_BOOLEAN	READ_WRITE	No
<b>ReferenceSubstraction:</b> Switches on or off the subtraction of the reference (background) spectra.	DEV_BOOLEAN	READ_WRITE	No
<b>TuneAtTimeIndex:</b> The calculated tune value from the tunes spectrum given by the attribute SpectrumIndex.	DEV_DOUBLE	READ	No
<b>TimerDelay:</b> Timer delay applied to the delay line in us	DEV_LONG	READ_WRITE	No
<b>TimeIndex:</b> The time index of the spectrum to be shown as profile in the array of spectra calculated. The time range is from 0 to 49 ms.	DEV_LONG	READ_WRITE	No

## Spectrum Attributes

Attribute name	Data Type	X Data Length	Expert
<b>Tunes:</b> The tune values calculated during an acceleration cycle of the booster.	DEV_DOUBLE	1000	No
<b>TuneSpectrum:</b> The calculated spectrum corresponding to the tune values given by the attribute SpectrumIndex.	DEV_DOUBLE	4096	No

## Image Attributes

Attribute name	Data Type	X Data Length	Y Data Length	Expert
<b>TuneSpectra:</b> All calculated spectra returned as an image.	DEV_DOUBLE	4096	1000	No

# Commands:

More Details on commands....

Device Commands for Operator Level		
Command name	Argument In	Argument Out
<b>Init</b>	DEV_VOID	DEV_VOID
<b>State</b>	DEV_VOID	DEV_STATE
<b>Status</b>	DEV_VOID	CONST_DEV_STRING
<b>StartAcquisition</b>	DEV_VOID	DEV_VOID
<b>StartReference</b>	DEV_VOID	DEV_VOID
<b>Stop</b>	DEV_VOID	DEV_VOID
<b>Reset</b>	DEV_VOID	DEV_VOID
<b>SaveReference</b>	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::STANDBY
  - 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::STANDBY
  - 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::STANDBY
  - Tango::FAULT
  - Tango::UNKNOWN

## 4 - StartAcquisition

- **Description:** Start the tune measurement. Switch on the shaker and start the data acquisition.
- **Argin:**  
**DEV\_VOID** :
- **Argout:**  
**DEV\_VOID** :

- **Command allowed for:**

- Tango::OFF
- Tango::STANDBY

## 5 - StartReference

- **Description:** Start the reference acquisition. Switch off the shaker and start the data acquisition with an averaging set to 10.

- **Argin:**  
DEV\_VOID :

- **Argout:**  
DEV\_VOID :

- **Command allowed for:**

- Tango::OFF

## 6 - Stop

- **Description:** Stop the tune measurement. Switch off the shaker and stop the data acquisition.

- **Argin:**  
DEV\_VOID :

- **Argout:**  
DEV\_VOID :

- **Command allowed for:**

- Tango::ON
- Tango::STANDBY
- Tango::FAULT
- Tango::UNKNOWN

## 7 - Reset

- **Description:** Reset the tune measurement after a detected fault.

- **Argin:**  
DEV\_VOID :

- **Argout:**  
DEV\_VOID :

- **Command allowed for:**

- Tango::OFF
- Tango::FAULT
- Tango::UNKNOWN

## 8 - SaveReference

- **Description:** Keep the actual tune spectra as reference for the reference (background) subtraction.  
The spectra are only kept in the server and not saved to disk. They are lost when restarting the server.
- **Argin:**  
**DEV\_VOID :**
- **Argout:**  
**DEV\_VOID :**
- **Command allowed for:**
  - Tango::STANDBY

TANGO is an open source project hosted by :  
**SOURCEFORGE.NET**<sup>®</sup>

Core and Tools : CVS repository on tango-cs project  
Device Servers : CVS repository on tango-ds project