



TANGO
Device
Server

Lakeshore_336 User's Guide

Lakeshore_336 Class

Revision: release_1_0_5 - Author: olivierroux
Implemented in C++ - CVS repository: tango-ds

Introduction:

basic interface for the Lakeshore Model 336 Temperature controller, GPIB/Ethernet interface.
reads 4 temperatures, outputs, alarms.

Class Inheritance:

- Tango::Device_4Impl
 - Lakeshore_336

Properties:

Device Properties		
Property name	Property type	Description
ComDeviceProxy	Tango::DEV_STRING	the Tango name of the communication device proxy (Ethernet or GPIB)
TimeInDeadBand	Tango::DEV_LONG	if temperature stays in deadband around setpoint for TimeInDeadBand seconds, the regulation is considered as successfull (the setpoint is reached, the stability is correct) The state of the device goes from RUNNING (blue color) to STANDBY (yellow)
Period	Tango::DEV_SHORT	polling period of the HW for the temperature and HW status
LoopNumber	Tango::DEV_SHORT	Loop number. possible values : 1 or 2. Default : 1

Device Properties Default Values:

Property Name	Default Values
ComDeviceProxy	No default value
TimeInDeadBand	60
Period	No default value
LoopNumber	No default value

There is no Class properties.

States:

States	
Names	Descriptions
ON	everything ok
UNKNOWN	communication broken
FAULT	parameterisation or blocking fault
ALARM	at least 1 relay is ON
MOVING	the controller tries to reach the set point (this state is used by scanserver to control the scans)
STANDBY	the controller has reached the setpoint (this state is used by ScanServer to control the scan)

Attributes:

Scalar Attributes			
Attribute name	Data Type	R/W Type	Expert
temperature: temperature voie principale en Kelvin read : temperature courante write : nouvelle consigne Attribut polle pour controler si la consigne est atteinte polling par default : 250 ms (si on passe 3 fois par consigne = temperature sans depassement d'1 Delta Temperature)	DEV_DOUBLE	READ_WRITE	No
output: output value of regulation from 0 to 100 %	DEV_DOUBLE	READ	Yes
range: current range can be Low(0.01) Medium(0.1) High(1)	DEV_SHORT	READ_WRITE	Yes
deadBand: dead band in witch we consider that setpoint is reached : if dead band is 0.5 e.g. if setpoint is 12 i;½K and temp is 12.4 -> OK if setpoint is 12 i;½K and temp is 11.6 -> OK if setpoint is 12 i;½K and temp is 11.4 -> KO	DEV_DOUBLE	READ_WRITE	Yes
inputA: temperature in Kelvin for Input A	DEV_DOUBLE	READ	No
inputB: temperature in Kelvin for Input B	DEV_DOUBLE	READ	No
inputC: temperature in Kelvin for Input C	DEV_DOUBLE	READ	No
inputD: temperature in Kelvin for Input D	DEV_DOUBLE	READ	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
Off	DEV_VOID	DEV_VOID
Low	DEV_VOID	DEV_VOID
Medium	DEV_VOID	DEV_VOID
High	DEV_VOID	DEV_VOID
LoopSelectInput	DEV_SHORT	DEV_VOID
IORaw	DEV_STRING	DEV_STRING
Stop	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::UNKNOWN
 - Tango::FAULT
 - Tango::ALARM
 - Tango::MOVING
 - Tango::STANDBY

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::UNKNOWN
 - Tango::FAULT
 - Tango::ALARM
 - Tango::MOVING
 - Tango::STANDBY

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::UNKNOWN
 - Tango::FAULT
 - Tango::ALARM
 - Tango::MOVING
 - Tango::STANDBY

4 - Off

- **Description:** turns OFF the temperature regulation : range set to 0
- **Argin:**
DEV_VOID :
- **Argout:**
DEV_VOID :
- **Command allowed for:**
 - Tango::ON
 - Tango::UNKNOWN
 - Tango::FAULT
 - Tango::ALARM
 - Tango::MOVING
 - Tango::STANDBY

5 - Low

- **Description:** turns ON the temperature regulation the range is set to 1 (1%, 0.5W max)
- **Argin:**
DEV_VOID :
- **Argout:**
DEV_VOID :

- **Command allowed for:**

- Tango::ON
- Tango::UNKNOWN
- Tango::FAULT
- Tango::ALARM
- Tango::MOVING
- Tango::STANDBY

6 - Medium

- **Description:** turns the temperature regulation ON the range is set to 2 (10%, 5W max)

- **Argin:**

DEV_VOID :

- **Argout:**

DEV_VOID :

- **Command allowed for:**

- Tango::ON
- Tango::UNKNOWN
- Tango::FAULT
- Tango::ALARM
- Tango::MOVING
- Tango::STANDBY

7 - High

- **Description:** turns the temperature regulation ON the range is set to 3 (100%, 50W max)

- **Argin:**

DEV_VOID :

- **Argout:**

DEV_VOID :

- **Command allowed for:**

- Tango::ON
- Tango::UNKNOWN
- Tango::FAULT
- Tango::ALARM
- Tango::MOVING
- Tango::STANDBY

8 - LoopSelectInput

- **Description:** selects the input for this loop
- **Argin:**
DEV_SHORT : input to associate with the loop [1|2|3|4]
- **Argout:**
DEV_VOID :
- **Command allowed for:**
 - Tango::ON
 - Tango::UNKNOWN
 - Tango::FAULT
 - Tango::ALARM
 - Tango::MOVING
 - Tango::STANDBY

9 - IORaw

- **Description:** sends a command (if no "?") requests a value (if "?")
- **Argin:**
DEV_STRING :
- **Argout:**
DEV_STRING :
- **Command allowed for:**
 - Tango::ON
 - Tango::UNKNOWN
 - Tango::FAULT
 - Tango::ALARM
 - Tango::MOVING
 - Tango::STANDBY

10 - Stop

- **Description:** tries to stop the regulation at the current temperature * sets the setpoint to the current temperature * stops the regulation manager (the state returns to STANDBY)
- **Argin:**
DEV_VOID :
- **Argout:**
DEV_VOID :

● **Command allowed for:**

- Tango::ON
- Tango::UNKNOWN
- Tango::FAULT
- Tango::ALARM
- Tango::MOVING
- Tango::STANDBY

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