TANGO Device Server

User's Guide

Agilent4395a Class

Revision: Agilent4395a-Release_1_1 - Author: pascal_verdier
Implemented in C++ - CVS repository: tango-ds

Introduction:

This classe handle Agilent 4395a device.

Class Inheritance:

- Tango::Device_4Impl
  - Agilent4395a

Class Description:

Properties:

Device Properties		
Property name	**Property type**	**Description**
Gpib_device_name	Tango::DEV_STRING	This is the name of the device, exported by the gpibDeviceServer.
Switch_delay	Tango::DEV_USHORT	This is the time kept for Agilent4395a for mode switching. Network and Spectrum mode commutation can be very long. This propertie allows user to adjust this value. Note too short value, will result in incorrect attribute restore (Freq/Sweep ect lost). UNIT is second.

Device Properties Default Values:

Property Name	**Default Values**
Gpib_device_name	No default value
Switch_delay	No default value

There is no Class properties.

States:

States	
Names	**Descriptions**
ON	
MOVING	

Attributes:

Scalar Attributes

Attribute name	Data Type	R/W Type	Expert
SweepStartFrequency: This is the starting frequency of the sweep.	DEV_DOUBLE	READ_WRITE	No
SweepStopFrequency: This is the stop frequency for the sweep.	DEV_DOUBLE	READ_WRITE	No
SweepTime: The sweep is done in the delay. Not that depending on the analyzer mode, you can't write SweepTime but SweepStep (spectrum mode) or you can write both attributes (network mode). Not that the device will adapt incorrect values to its capacities.	DEV_DOUBLE	READ_WRITE	No
Voltage: Output level in Network mode (spectrum does not generate output signal).	DEV_DOUBLE	READ_WRITE	No
TriggerSource: This attricbute control what's triggering a measurement. IMM is restarting at the end of the sweep. BUS is a one shot mesure triggered by gpib bus. EXT is a one shot measurement triggered by an external device/signal.	DEV_STRING	READ_WRITE	No
SweepStep: The sweep is done by increasing frequency with sweep step until SweepStopFrequency is reached.	DEV_DOUBLE	READ_WRITE	No
ReferenceLevel: This attribute center display on the given value. Note that this attribute only affect device's screen.	DEV_DOUBLE	READ_WRITE	No
Scale: This is the nbr of dbm represented by one unit on the device's screen.	DEV_DOUBLE	READ_WRITE	No
AttenuatorA: This is the attenuation of the input port A. Allowed values are 0, 10, 20, 30, 40, 50 DB.	DEV_SHORT	READ_WRITE	No
AttenuatorB: This is the attenuation of the input port B. Allowed values are 0, 10, 20, 30, 40, 50 DB.	DEV_SHORT	READ_WRITE	No
AttenuatorR: This is the attenuation of the input port R. Allowed values are 0, 10, 20, 30, 40, 50 DB.	DEV_SHORT	READ_WRITE	No
NAverage: This attibute determines on how many sweep averaging is done.	DEV_SHORT	READ_WRITE	No
Averaging: This attribute represent averaging state: on or off.	DEV_BOOLEAN	READ_WRITE	No
Measure: What measure is done: A (spectrum) or A/R, B/R (network).	DEV_STRING	READ_WRITE	No
Mode: This mode selects between spectrum mode and Network mode.	DEV_STRING	READ_WRITE	No
DataAcquisition: This attribute enable / disable data acquistion. Technicaly it send HOLD/CONT command to the 4395a.	DEV_BOOLEAN	READ_WRITE	No

Spectrum Attributes

Attribute name	Data Type	X Data Length	Expert
Spectrum: This is the spectrum returned by the device after sweep and analyzes.	DEV_DOUBLE	800	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

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::MOVING

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::MOVING

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::MOVING