

## Modbus Tango Cpp Class

### Contents :

- [Description](#)
- [Properties](#)
- [Commands](#)
  - [State](#)
  - [Status](#)
  - [ForceSingleCoil](#)
  - [ReadCoilStatus](#)
  - [ReadInputStatus](#)
  - [ReadHoldingRegisters](#)
  - [ReadInputRegisters](#)
  - [PresetSingleRegister](#)
  - [ReadExceptionStatus](#)
  - [FetchCommEventCtr](#)
  - [ForceMultipleCoils](#)
  - [ReadMultipleCoilsStatus](#)
  - [PresetMultipleRegisters](#)
  - [MaskWriteRegister](#)
  - [ReadWriteRegister](#)
  - [PresetSingleRegisterBroadcast](#)
- [Attributes](#)
- [States](#)

### Modbus Class Identification :

Contact : at esrf.fr - accelerator-control  
Class Family : Communication  
Platform : All Platforms  
Bus : Modbus  
Manufacturer : none  
Manufacturer ref. :

### Modbus Class Inheritance :

- [Tango::DeviceImpl](#)
  - Modbus

### Modbus Class Description :

A Class to handle the modbus protocol over TCP/IP or Serial (RTU).

## Modbus Properties :

### There is no class properties

Device Properties			
Name	Description	Type	Default Value
Protocol	RTU`` : Binary serial communication. ``TCP`` : Communication over ethernet.	String	RTU
lphost	The host IP address used with the TCP protocol in the form aa.bb.cc.dd.	String	none
Serialline	The name of the serial line device used with RTU protocol. This can be any device name of a Serial Class object in the Tango system.	String	none
Address	Node index used with the RTU or TCP protocol	short	1
CacheConfig	Describe which data has to be cached. Each set of cached data is described by 3 parameters which are: 1 - Command to be used to read data (ReadHoldingRegisters, ReadInputStatus ReadInutRegisters or ReadMultipleCoilStatus) 2 - First address to be read 3 - Number of data to read	String[]	none
CacheSleep	Cache update thread main loop sleeping time (in ms)	int	1000
SocketConnectionSleep	The necessary sleep time between closing a connection (Socket) and opening a new connection. To avoid hang-ups a non blocking socket is used to check the availability on the network. Afterwards the non blocking socket is closed and a blocking socket will be opened. The SocketConnectionSleep time specified the wait time in ms between these two connections.	int	200
TCPTimeout	Timeout used when the TCP protocol is used (in sec)	double	1
LogFile	Name og the file where are stored invalid frame	String	none
TCPNoDelay	Disable Nagle`s algorithm. Set this property to reduce the latency of TCP transmissions. This property is relevant only if the ``Protocol`` property is set to ``TCP``.	boolean	false
TCPQuickAck	Set this property to true to enable TCP quick acknowledgements to reduce the TCP latency. This could be useful with some electronic boards to force the device server to acknowledge immediately the received TCP packets. This avoids TCP packets retransmissions and reduces the TCP latency. Please note that this is relevant only if ``Protocol`` property is set to ``TCP``. Please also note that this works only on Linux Operating Systems.	boolean	false

Modbus Class Commands				
Name	Input type	Output type	Level	Description
<a href="#">State</a>	DEV_VOID	DEV_STATE	OPERATOR	This command gets the device state (stored in its <i>device_state</i> data member) and returns it to the caller.
<a href="#">Status</a>	DEV_VOID	CONST_DEV_STRING	OPERATOR	This command gets the device status (stored in its <i>device_status</i> data member) and returns it to the caller.
<a href="#">ForceSingleCoil</a>	DEVVAR_SHORTARRAY	DEV_VOID	OPERATOR	Write single coil (digital I/O) state.
<a href="#">ReadCoilStatus</a>	DEV_SHORT	DEV_SHORT	OPERATOR	Read coil (digital I/O) status.
<a href="#">ReadInputStatus</a>	DEVVAR_SHORTARRAY	DEVVAR_CHARARRAY	OPERATOR	Read discrete input status. Return one boolean per array element.
<a href="#">ReadHoldingRegisters</a>	DEVVAR_SHORTARRAY	DEVVAR_SHORTARRAY	OPERATOR	Read multiple 16bits registers.
<a href="#">ReadInputRegisters</a>	DEVVAR_SHORTARRAY	DEVVAR_SHORTARRAY	OPERATOR	Read Multiple 16bits input registers.
<a href="#">PresetSingleRegister</a>	DEVVAR_SHORTARRAY	DEV_VOID	OPERATOR	Write single 16bits register.
<a href="#">ReadExceptionStatus</a>	DEV_VOID	DEV_SHORT	OPERATOR	Read exception status (usually a predefined range of 8 bits)
<a href="#">FetchCommEventCtr</a>	DEV_VOID	DEVVAR_SHORTARRAY	OPERATOR	Fetch communications event counter.
<a href="#">ForceMultipleCoils</a>	DEVVAR_SHORTARRAY	DEV_VOID	OPERATOR	Write multiple coils (digital I/O) state. argin[0] = coil_address argin[1] = number of coils argin[2] = 1st coil state argin[3] = 2nd coil state ...
				Read multiple

<a href="#">ReadMultipleCoilsStatus</a>	DEVVAR_SHORTARRAY	DEVVAR_SHORTARRAY	OPERATOR	coil (digital I/O) status. argin[0] = register address argin[1] = number of registers
<a href="#">PresetMultipleRegisters</a>	DEVVAR_SHORTARRAY	DEV_VOID	OPERATOR	Write multiple 16bits registers. argin[0] = register address argin[1] = number of registers argin[2] = 1st register argin[3] = 2nd register ...
<a href="#">MaskWriteRegister</a>	DEVVAR_SHORTARRAY	DEV_VOID	OPERATOR	Mask write a 16bits register.
<a href="#">ReadWriteRegister</a>	DEVVAR_SHORTARRAY	DEVVAR_SHORTARRAY	OPERATOR	Read and Write multiple 16bits registers. argin[0] = read address argin[1] = nb of registers to read argin[2] = write address, argin[3] = nb of registers to write, argin[4] = 1st register value to write argin[5] = 2nd register value to write ...
<a href="#">PresetSingleRegisterBroadcast</a>	DEVVAR_SHORTARRAY	DEV_VOID	OPERATOR	Write single 16bits register at address 0 (Address reserved for broadcast) Does not wait for the equipment response.

**Command State :**

This command gets the device state (stored in its *device\_state* data member) and returns it to the caller.

State Definition		
Input Argument	Tango::DEV_VOID	none.
Output Argument	Tango::DEV_STATE	State Code
DisplayLevel	OPERATOR	..
Inherited	true	..
Abstract	true	..
Polling Period	Not polled	..
Command allowed for	All states	..

---

### **Command Status :**

This command gets the device status (stored in its *device\_status* data member) and returns it to the caller.

Status Definition		
Input Argument	Tango::DEV_VOID	none.
Output Argument	Tango::CONST_DEV_STRING	Status description
DisplayLevel	OPERATOR	..
Inherited	true	..
Abstract	true	..
Polling Period	Not polled	..
Command allowed for	All states	..

---

### **Command ForceSingleCoil :**

Write single coil (digital I/O) state.

ForceSingleCoil Definition		
Input Argument	Tango::DEVVAR_SHORTARRAY	coil address, 0/1
Output Argument	Tango::DEV_VOID	

DisplayLevel	OPERATOR	..
Inherited	false	..
Abstract	false	..
Polling Period	Not polled	..
Command allowed for	All states	..

### **Command ReadCoilStatus :**

Read coil (digital I/O) status.

<b>ReadCoilStatus Definition</b>		
Input Argument	Tango::DEV_SHORT	coil address
Output Argument	Tango::DEV_SHORT	Coil status
DisplayLevel	OPERATOR	..
Inherited	false	..
Abstract	false	..
Polling Period	Not polled	..
Command allowed for	All states	..

### **Command ReadInputStatus :**

Read discrete input status. Return one boolean per array element.

<b>ReadInputStatus Definition</b>		
Input Argument	Tango::DEVVAR_SHORTARRAY	input address, no. of inputs
Output Argument	Tango::DEVVAR_CHARARRAY	Input status.
DisplayLevel	OPERATOR	..
Inherited	false	..
Abstract	false	..
Polling Period	Not polled	..
Command allowed for	All states	..

---

**Command ReadHoldingRegisters :**

Read multiple 16bits registers.

<b>ReadHoldingRegisters Definition</b>		
Input Argument	Tango::DEVVAR_SHORTARRAY	register address, no. of registers
Output Argument	Tango::DEVVAR_SHORTARRAY	Holding 16bits register.
DisplayLevel	OPERATOR	..
Inherited	false	..
Abstract	false	..
Polling Period	Not polled	..
Command allowed for	All states	..

---

**Command ReadInputRegisters :**

Read Multiple 16bits input registers.

<b>ReadInputRegisters Definition</b>		
Input Argument	Tango::DEVVAR_SHORTARRAY	register address, no. of registers
Output Argument	Tango::DEVVAR_SHORTARRAY	Input 16bits registers
DisplayLevel	OPERATOR	..
Inherited	false	..
Abstract	false	..
Polling Period	Not polled	..
Command allowed for	All states	..

---

**Command PresetSingleRegister :**

Write single 16bits register.

<b>PresetSingleRegister Definition</b>		
Input Argument	Tango::DEVVAR_SHORTARRAY	Register address, register value.
Output Argument	Tango::DEV_VOID	
DisplayLevel	OPERATOR	..
Inherited	false	..
Abstract	false	..
Polling Period	Not polled	..
Command allowed for	All states	..

### **Command ReadExceptionStatus :**

Read exception status (usually a predefined range of 8 bits)

<b>ReadExceptionStatus Definition</b>		
Input Argument	Tango::DEV_VOID	
Output Argument	Tango::DEV_SHORT	exception status
DisplayLevel	OPERATOR	..
Inherited	false	..
Abstract	false	..
Polling Period	Not polled	..
Command allowed for	All states	..

### **Command FetchCommEventCtr :**

Fetch communications event counter.

<b>FetchCommEventCtr Definition</b>		
Input Argument	Tango::DEV_VOID	
Output Argument	Tango::DEVVAR_SHORTARRAY	status, event count
DisplayLevel	OPERATOR	..
Inherited	false	..

Abstract	false	..
Polling Period	Not polled	..
Command allowed for	All states	..

### **Command ForceMultipleCoils :**

Write multiple coils (digital I/O) state.

argin[0] = coil\_address

argin[1] = number of coils

argin[2] = 1st coil state

argin[3] = 2nd coil state

...

<b>ForceMultipleCoils Definition</b>		
Input Argument	Tango::DEVVAR_SHORTARRAY	coil address, nb of coils, coil states
Output Argument	Tango::DEV_VOID	
DisplayLevel	OPERATOR	..
Inherited	false	..
Abstract	false	..
Polling Period	Not polled	..
Command allowed for	All states	..

### **Command ReadMultipleCoilsStatus :**

Read multiple coil (digital I/O) status.

argin[0] = register address

argin[1] = number of registers

<b>ReadMultipleCoilsStatus Definition</b>		
Input Argument	Tango::DEVVAR_SHORTARRAY	coil address, nb of coils
Output Argument	Tango::DEVVAR_SHORTARRAY	Status of coils
DisplayLevel	OPERATOR	..
Inherited	false	..
Abstract	false	..
Polling Period	Not polled	..

Command allowed for	All states	..
---------------------	------------	----

### **Command PresetMultipleRegisters :**

Write multiple 16bits registers.  
 argin[0] = register address  
 argin[1] = number of registers  
 argin[2] = 1st register  
 argin[3] = 2nd register  
 ...

<b>PresetMultipleRegisters Definition</b>		
Input Argument	Tango::DEVVAR_SHORTARRAY	register address, nb of registers, register data
Output Argument	Tango::DEV_VOID	
DisplayLevel	OPERATOR	..
Inherited	false	..
Abstract	false	..
Polling Period	Not polled	..
Command allowed for	All states	..

### **Command MaskWriteRegister :**

Mask write a 16bits register.

<b>MaskWriteRegister Definition</b>		
Input Argument	Tango::DEVVAR_SHORTARRAY	register address, AND mask, OR mask
Output Argument	Tango::DEV_VOID	
DisplayLevel	OPERATOR	..
Inherited	false	..
Abstract	false	..
Polling Period	Not polled	..
Command allowed for	All states	..

---

### **Command ReadWriteRegister :**

Read and Write multiple 16bits registers.

argin[0] = read address

argin[1] = nb of registers to read

argin[2] = write address,

argin[3] = nb of registers to write,

argin[4] = 1st register value to write

argin[5] = 2nd register value to write

...

<b>ReadWriteRegister Definition</b>		
Input Argument	Tango::DEVVAR_SHORTARRAY	read address, no. to read, write address, nb.of write, write data
Output Argument	Tango::DEVVAR_SHORTARRAY	read registers
DisplayLevel	OPERATOR	..
Inherited	false	..
Abstract	false	..
Polling Period	Not polled	..
Command allowed for	All states	..

---

### **Command PresetSingleRegisterBroadcast :**

Write single 16bits register at address 0 (Address reserved for broadcast)

Does not wait for the equipment response.

<b>PresetSingleRegisterBroadcast Definition</b>		
Input Argument	Tango::DEVVAR_SHORTARRAY	register value.
Output Argument	Tango::DEV_VOID	
DisplayLevel	OPERATOR	..
Inherited	false	..
Abstract	false	..
Polling Period	Not polled	..
Command allowed for	All states	..

---

---

**There is no attribute defined.**

---

**There is no dynamic attribute defined.**

---

---

**There is no state defined**