Station Automation COM600 3.5
Modbus TCP Slave (OPC) User's Manual

Power and productivity
for a better world™

Contents:

1. About this manual

1.1. Copyrights

1.2. Trademarks

1.3. General

This manual provides thorough information on the and the central concepts related to it. You find instructions on how to take it into use. The basic operation procedures are also discussed.

Information in this user’s manual is intended for application engineers who configure to establish data transfer between the Modbus master system and process devices connected to COM600.

As a prerequisite, you should understand the Modbus protocol and the basic procedures in Station Automation Builder 600 (later referred to as SAB600).

This user’s manual is divided into following sections:

Introduction

Configuration

In this section you find an overview of the configuration tasks and instructions on how to create and configure related objects.

Operation

This section covers the basic operation procedures you can carry out when transferring or activating Station Automation COM600 (later referred to as COM600) with new configurations.

You are also given instructions on how to monitor and control the Modbus communication.

Technical reference

This section describes the IEC 61850 data modeling, contains attributes and a list of status codes.

1.4. Document conventions

The following conventions are used for the presentation of material:

- The words in names of screen elements (for example, the title in the title bar of a window, the label for a field of a dialog box) are initially capitalized.
- Capital letters are used for the name of a keyboard key if it is labeled on the keyboard. For example, press the ENTER key.
- Lowercase letters are used for the name of a keyboard key that is not labeled on the keyboard. For example, the space bar, comma key, and so on.
- Press CTRL+C indicates that you must hold down the CTRL key while pressing the C key (to copy a selected object in this case).
- Press ESC E C indicates that you press and release each key in sequence (to copy a selected object in this case).
- The names of push and toggle buttons are boldfaced. For example, click **OK**.
- The names of menus and menu items are boldfaced. For example, the **File** menu.
  - The following convention is used for menu operations: **MenuName** > **MenuItem** > **CascadedMenuItem**. For example: select **File** > **New** > **Type**.
  - The **Start** menu name always refers to the **Start** menu on the Windows taskbar.
- System prompts/messages and user responses/input are shown in the Courier font. For example, if you enter a value out of range, the following message is displayed:

```
Entered value is not valid. The value must be 0 - 30  .
```

- You can be asked to enter the string MIF349 in a field. The string is shown as follows in the procedure:

  MIF349
- Variables are shown using lowercase letters:

  sequence name

1.5. Use of symbols

This publication includes warning, caution, and information icons that point out safety-related conditions or other important information. It also includes tip icons to point out useful information to the reader. The corresponding icons should be interpreted as follows.

The electrical warning icon indicates the presence of a hazard which could result in electrical shock.

The warning icon indicates the presence of a hazard which could result in personal injury.

The caution icon indicates important information or warning related to the concept discussed in the text. It may indicate the presence of a hazard which could result in corruption of software or damage to equipment or property.

The information icon alerts the reader to relevant facts and conditions.

The tip icon indicates advice on, for example, how to design your project or how to use a certain function.

1.6. Terminology

The following is a list of terms associated with COM600 that you should be familiar with. The list contains terms that are unique to ABB or have a usage or definition that is different from standard industry usage.

Term	Description
Alarm	An abnormal state of a condition.

Term	Description
Alarms and Events; AE	An OPC service for providing information about alarms and events to OPC clients.
Data Access; DA	An OPC service for providing information about process data to OPC clients.
Data Object; DO	Part of a logical node object representing specific information, for example, status, or measurement. From an object-oriented point of view, a data object is an instance of a class data object. DOs are normally used as transaction objects; that is, they are data structures.
Data Set	The data set is the content basis for reporting and logging. The data set contains references to the data and data attribute values.
Device	A physical device that behaves as its own communication node in the network, for example, protection relay.
Event	Change of process data or an OPC internal value. Normally, an event consists of value, quality, and timestamp.
Intelligent Electronic Device	A physical IEC 61850 device that behaves as its own communication node in the IEC 61850 protocol.
Logical Device; LD	Representation of a group of functions. Each function is defined as a logical node. A physical device consists of one or several LDs.
Logical Node; LN	The smallest part of a function that exchanges data. An LN is an object defined by its data and methods.
LON	A communication protocol developed by Echelon.
LON Application Guideline for substation automation; LAG	A proprietary method of ABB on top of the standard LON protocol.
OPC	Series of standards specifications aiming at open connectivity in industrial automation and the enterprise systems that support industry.
OPC item	Representation of a connection to the data source within the OPC server. An OPC item is identified by a string <object path>:<property name>. Associated with each OPC item are Value, Quality, and Time Stamp.
Property	Named data item.
Report Control Block	The report control block controls the reporting processes for event data as they occur. The reporting process continues as long as the communication is available.
SPA	ABB proprietary communication protocol used in substation automation.
SPA device	Protection and/or Control Product supporting the SPA protocol version 2.5 or earlier.
Substation Configuration Language; SCL	XML-based description language for configurations of electrical substation IEDs. Defined in IEC 61850 standard.

1.7. Abbreviations

The following is a list of abbreviations associated with COM600 that you should be familiar with. See also 1.6, Terminology.

Abbreviation	Description
AE	Alarms and Events
ASDU	Application Service Data Unit
BRCB	Buffered Report Control Block
DA	Data Access
DMCD	Data Message Code Definition
DO	Data Object
GW	Gateway, component connecting two communication networks together
HMI	Human Machine Interface
IEC	International Electrotechnical Commission
IED	Intelligent Electronic Device
LAG	LON Application Guideline for substation automation
LAN	Local Area Network
LD	Logical Device
LMK	LonMark interoperable device communicating in LonWorks network. In this document, the term is used for devices that do not support the ABB LON/LAG communication.
LN	Logical Node
LSG	LON SPA Gateway
NCC	Network Control Center
NUC	Norwegian User Convention
NV	Network Variable
OLE	Object Linking and Embedding
OPC	OLE for Process Control
P&C	Protection & Control
RTS	Request To Send
SA	Substation Automation
SAB600	Station Automation Builder 600
SCL	Substation Configuration Language
SLD	Single Line Diagram
SNMP	Simple Network Management Protocol
SNTP	Simple Network Time Protocol

Abbreviation	Description
SOAP	Simple Object Access Protocol
RCB	Report Control Block
URCB	Unbuffered Report Control Block
XML	eXtended Markup Language

1.8. Related documents

Name of the manual	MRS number
COM600 User’s Manual	1MRS756125

1.9. Document revisions

Document version/date	Product revision	History
A/06.11.2009	3.4	Document created
B/30.06.2011	3.5	Document revised

2. Introduction

2.1. Functional overview

The Modbus protocol slave interface of COM600 enables master systems communicating with Modbus protocol to:
- receive data from P&C devices
- deliver commands to P&C devices connected to the Gateway.

The Modbus slave is implemented as an OPC client, which transfers and converts data between the Modbus slave protocol interface and the OPC servers of COM600. For more information, see COM600 User's Manual.

The Modbus TCP Slave OPC Client is configured using SAB600. SAB600 can also be used for diagnosing and controlling the operation of the Modbus TCP Slave OPC Client. COM600 has a web server that can be used for remote diagnostic of the Gateway including the Modbus TCP Slave OPC Client.

The Modbus TCP Slave OPC Client uses TCP/IP communication over the TCP interface. Before you can start using the Modbus TCP Slave OPC Client, configure at least one OPC server to provide access to the process devices. In this user's manual, the term "Modbus TCP IED" is used for a virtual station in COM600 representing the slave stations visible to the Modbus master system.

2.2. Modbus TCP Slave OPC Client features

The Modbus TCP Slave OPC Client supports the following features:
- OPC Data Access Client v. 1.0/2.0 for accessing data from the OPC servers
- OPC Alarms and Events specifications v. 1.10 for diagnostic and control purposes
- IEC 61850 data modeling
- System supervision:
  - NCC connection supervision

Table 2.2-1 Supported function codes

Function Code	Description	Memory Area
01	Read Coil Status	00001 - 09999
03	Read Holding Register	40001 - 49999
05	Force Single Coil	00001 – 09999
06	Write Single Register	40001 – 49999
16	Write Multiple Registers	40001 - 49999

Supported data formats:
- Bit - one coil status.

- Word - one register. The data is used in an unsigned form.
- Integer - one register. The MSB bit is used as a sign bit.
- Long MSW last - signed 32-bit object, which needs two registers in LSW-MSW order.
- Long MSW first - signed 32-bit object, which needs two registers in MSW-LSW order.
- Float MSW last - floating point type, which needs two registers in LSW-MSW order.
- Float MSW first - floating point type, which needs two registers in MSW-LSW order.

3. Configuration

3.1. About this section

This section guides you in the configuration tasks required before you can start using the Modbus TCP Slave OPC Client. For information on the IEC 61850 data modeling, see COM600 User's Manual.

Start Station Automation Builder 600 (later referred to as SAB600) to open a project where at least one OPC server has been configured. You can also open and name a new project, where you configure at least one OPC server.

1. Select **File > Open/Manage Project...**.
2. In the Open/Manage Project dialog, select the required location for the project:
   - Projects on my computer
   - Projects on the network
3. Select **New Project** on the left.
   - Enter a project name. The description is optional.
4. Click **Create**.
5. Click **Open Project**.

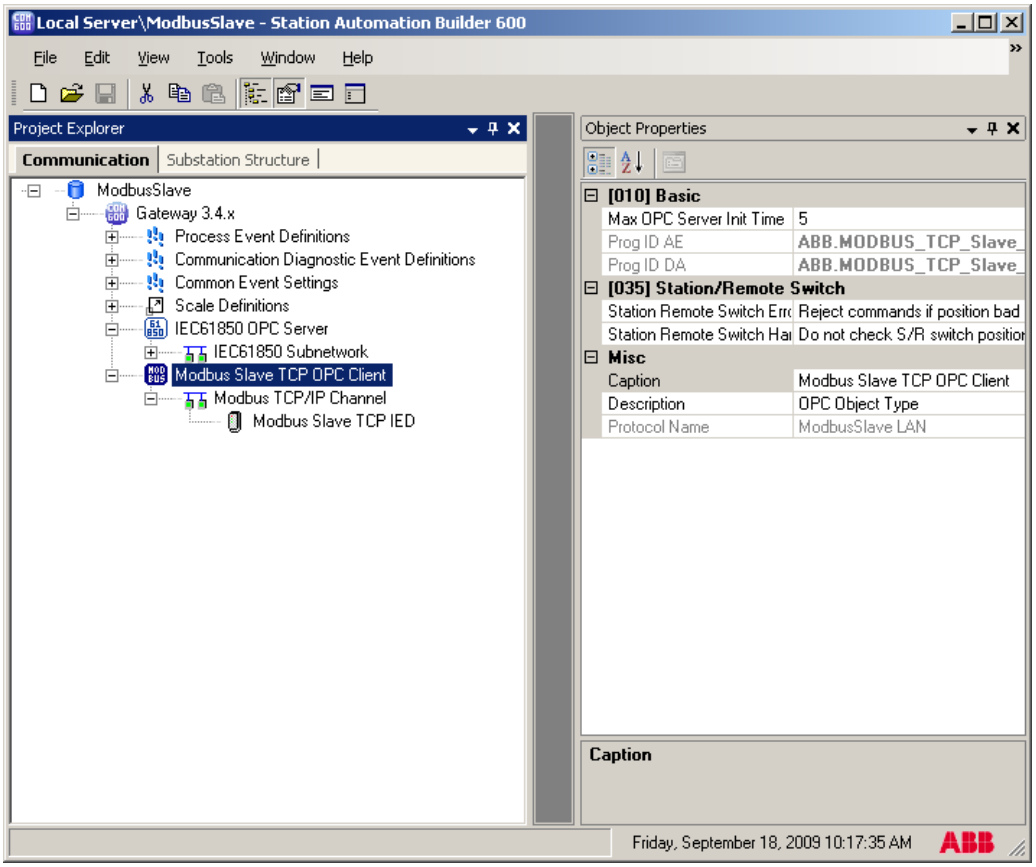
3.2. Overview of configuration

Before you can start using the Modbus TCP Slave OPC Client, build and configure an object tree in SAB600 to define the Communication structure within the Gateway object.

- Modbus TCP Slave OPC Client
- Modbus TCP Channel
- Modbus TCP IED
- Data objects

Figure 3.2-1 shows an example view of SAB600 including an object tree in the communication structure on the left and Object Properties window displaying the object properties on the right.

When configuring OPC servers the following characters cannot be used in object names: \ ` ' ' #



example_view_SAB_TCP.bmp

Figure 3.2-1 Example view of SAB600

To configure an object tree:

1. Build an object tree by adding the necessary objects to the object tree, see 3.3.1, General about building object tree and 3.3.5, Adding data objects using Cross-References function.
   Figure 3.2-1 shows an example of how the object tree may look like after it has been built. In the example tree you can see the Modbus TCP Slave OPC Client object and its child objects, such as channels, devices, and data objects. Indentation is used to indicate the parent-child relationship between the objects.
2. Configure the object properties in the communication structure, see 3.4.1, General about configuring objects.

The following table describes the objects shown in the object tree (Figure 3.2-1).

Table 3.2-1 Modbus TCP Slave OPC Client related objects

Object	Description
Modbus TCP Slave OPC Client	An object representing the Modbus TCP Slave OPC Client.
Modbus TCP Channel	An object representing the channel.

Object	Description
Modbus TCP IED	A Modbus TCP IED is used for a virtual station in COM600 representing the slave stations visible to the Modbus master system.
Data Object (DO)	A data object is an instance of one of the IEC Common data classes, for example single point status, measured value etc. Depending on the class, each data object has a set of attributes for monitoring and controlling the object, for instance value, quality, and control. Data objects are connected from OPC servers to the Modbus TCP Slave OPC Client with the cross-reference function. They are shown as child objects of the Modbus TCP IED object in the object tree.
Event Definitions	Event definitions are used for the diagnostic OPC A&E Server.

3.3. Building object tree

3.3.1. General about building object tree

The object tree is built in the communication structure of SAB600 by adding objects in a logical order starting from the Modbus TCP Slave OPC Client object. For more information, see Figure 3.2-1.

Before the Modbus TCP Slave OPC Client can be taken into use, configure an OPC server for the process communication. For more information on creating an OPC server, see COM600 User's Manual.

To add objects to the object tree in the communication structure:
- right-click the object to which you want to add a child object, or
- copy the object.

Add the objects in the following order:
1. Modbus TCP Slave OPC Client
2. Modbus TCP Channel
3. Modbus TCP IED
4. Data objects.

3.3.2. Adding Modbus TCP Slave OPC Client object

To add the OPC client object in the communication structure:
1. Select the gateway object.
2. Right-click the gateway object and select **New** > **Modbus** > **Modbus TCP Slave OPC Client**.

3.3.3. Adding channel objects

After the Modbus TCP Slave OPC Client object has been successfully added, continue building the object tree by adding a Modbus TCP Channel object.

To add Modbus TCP Channel object:
1. Select a Modbus TCP Slave OPC Client object and right-click it.
2. Add a Modbus TCP Channel object.
3. Rename the new object. The names of the Modbus TCP Channel objects within a Modbus TCP Slave OPC Client must be unique.

3.3.4. Adding Modbus TCP IED object

After a channel object has been successfully added, continue building the structure by adding the Modbus TCP Channel object. All the data can be connected to one device or divided to several slave devices. Before dividing data to several slave devices, check that the current protocol mode and the master system support the feature.

To add a Modbus TCP IED object:
1. Select a Modbus TCP Channel object.
2. Add a Modbus TCP IED object.
3. Rename the new object. The names within Modbus TCP Channel must be unique.

3.3.5. Adding data objects using Cross-References function

Data objects are added by dragging and dropping from an OPC server to the Modbus TCP Slave OPC Client.

To add data objects:
1. Select a Modbus TCP IED object and right-click it.
2. Select **Cross-References**. The Cross-References function appears (see Figure 3.3.5-1).
3. In the Project Explorer, select a logical node within an OPC server, from which you want to connect the data objects to the Modbus TCP Slave OPC Client.

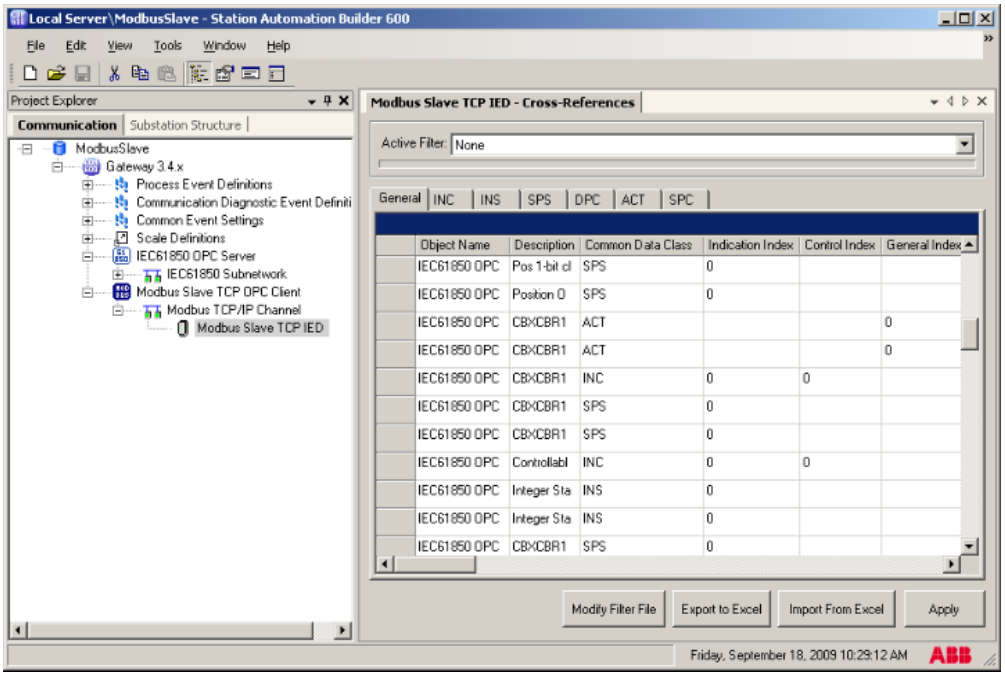
You can also select an upper level (server, channel, etc.) object and drag and drop it into the Cross-References function. All the data objects within the selected object appear in the Cross-References function and can be connected to the Modbus TCP Slave OPC Client.

4. Drag and drop the logical node into the Cross-References function. The data objects within the logical node appear in the Cross-References function.
5. Specify the addresses that map to the data objects.

Only Coils and Holding registers can be mapped, input status and input registers should not be used.

Only data objects that have been given a non-zero information address in the Cross-References table are connected to the Modbus TCP IED.

6. Click **Apply** to create the cross-references (to connect the data objects to the Modbus TCP IED).



Modbus_TCP_Slave_Cross_References.bmp

Figure 3.3.5-1 The Cross-References window

For more information about the Cross-References function, see COM600 User's Manual.

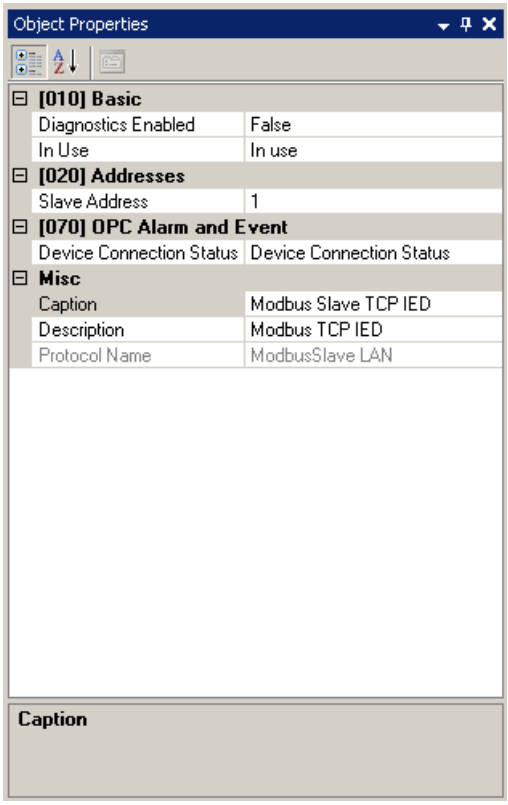
3.4. Configuring objects

3.4.1. General about configuring objects

After the objects have been added, configure the object properties. Figure 3.4.1-1 shows an example of how to use SAB600 to configure the object properties for Modbus TCP Slave OPC Client.

To configure an object:

1. Select an object in the object tree of the communication structure.
   The object properties appear now in the Object Properties window. The properties and their values can be viewed as shown in Figure 3.4.1-1.



example_object_properties.bmp

Figure 3.4.1-1 Example of object properties in the Objects Properties window

2. Select the property you want to configure.

   Depending on the property value type, configuring is done by:
   - selecting a predefined value from a drop-down menu, or
   - entering a text string or a numerical value into a text field.

The available properties for different objects are listed in the following subsections.

3.4.2. Configuring Modbus TCP Slave OPC Client properties

Table 3.4.2-1 lists the configurable Modbus TCP Slave OPC Client properties and their value ranges. The actual configuration by using SAB600 is performed as described in 3.2, Overview of configuration.

Table 3.4.2-1 Modbus TCP Slave OPC Client properties

Property/Parameter	Value or Value range/Default	Description
Basic		
Maximum OPC Server Initialization Time	0...65535 Default: 5	Specifies the maximum time in seconds that any connected (configured) OPC Server requires to retrieve all its initial data.
Prog ID AE		Instance identification of a diagnostic OPC alarm and event server.
Prog ID DA		Instance identification of a diagnostic OPC data access server.
Station/Remote Switch		
Station/Remote Switch Handling	Do not check Station/Remote switch position. Check Station/Remote switch position. Default: Do not check Station/Remote switch position.	Specifies if the position of the station remote switch is going to be checked.
Station/Remote Switch Error	Reject commands if position bad or unknown. Allow commands if position bad or unknown. Default: Reject commands if position bad or unknown.	Defines command handling, if the position is bad or unknown.

3.4.3. Configuring Modbus TCP Channel properties

The Modbus TCP Channel properties that can be configured and their value ranges are listed in Table 3.4.3-1. The actual configuration by using SAB600 is performed as described in 3.4.1, General about configuring objects.

Table 3.4.3-1 Modbus TCP Channel properties

Property/Parameter	Value or Value range/Default	Description
Basic		
In use	In use Not in use Default: In use	Specifies whether the channel is in use or not.

Property/Parameter	Value or Value range/Default	Description
Protocol	Modbus Slave over TCP interface protocol	
Communication Port		
Local Address	Default: 127.0.0.1	The locally used IP address.
Communication Port	Default: 502	The port that the server is listening on.

3.4.4. Configuring Modbus TCP IED properties

Table 3.4.4-1 lists the configurable properties for Modbus TCP IED and their value ranges. The actual configuration by using SAB600 is performed as described in 3.4.1, General about configuring objects.

Table 3.4.4-1 Modbus TCP IED properties

Name	Value/Value range	Description
Basic		
Diagnostics Enabled	True False Default: False	Specifies if diagnostic AE events are sent for the station.
In Use	In use Not in use Default: In use	Defines if the IED is in use or not.
Addresses		
Slave Address	0...255 Default: 1	The station address of the slave station.

3.4.5. Data object configuration

3.4.5.1. Configuring data objects

Configure data objects either in the **Object Properties** window or in the **Cross-References** window.

The actual configuration in the **Object Properties** window by using SAB600 is performed as described in 3.4.1, General about configuring objects.

To configure the data objects in the Cross-References window:
1. Select the IED object in the object tree and right-click it.

2. Select the **Cross-References** window from the context menu.
3. Change the values in cross-references table by entering the new value in the table cell with the desired property.
4. Click **Apply** to save the changes and to connect the data objects to the IED. The connected data objects appear as child objects for the IED. Modify the cross-reference information by selecting the data object and using the object properties window.

The parameters are stored in the object properties in SAB600 (see the tables for each data object type).

Table 3.4.5.1-1 Valid address ranges for configuring address values for Modbus IED data objects

Value range	Address value
00001..9999	Coils, 0X references
40001..49999	Holding registers, 4X references

Address value 0 means that the corresponding information is not available or not used in the configuration.

If you change the object names or the structure of objects connected to the IED in the OPC Server, open the cross-reference tool and verify that the changes are correctly handled. Click **Apply** to update the configuration accordingly.

Modbus OPC Client supports data objects for status, measurements, controllable status, and controllable analog information. The following subsections list the configurable data object properties for the Modbus OPC Client.

3.4.5.2. Directional protection activation information (ACD)

Table 3.4.5.2-1 Configurable ACD properties

Property/ Parameter	Value or Value range/ Default	Description
Basic		
Common Data Class	ACD	Common data class according to IEC 61850.
Addresses		
General Address	0...65535 Default 0	General Address.
Neutral Address	0...65535 Default 0	Neutral Address.

Property/ Parameter	Value or Value range/ Default	Description
Phase A Address	0...65535 Default 0	Phase A Address.
Phase B Address	0...65535 Default 0	Phase B Address.
Phase C Address	0...65535 Default 0	Phase C Address.
Common		
Update Rate	0...60000 Default: 0	Maximum update rate of indication changes between the OPC server and the client in milliseconds. The value 0 means that the server sends all the changes to the client.
Data Specific		
Send as Double Point	True False Default: False	Specifies if a value of an indication signal is sent as a double point value.
Send as Inverse Value	True False Default: False	Specifies if a value of an indication signal is sent as an inverse value.

3.4.5.3. Protection activation information (ACT)

Table 3.4.5.3-1 Configurable ACT properties

Property/ Parameter	Value or Value range/ Default	Description
Basic		
Common Data Class	ACT	Common data class according to IEC 61850.
Addresses		
General Address	0...65535 Default: 0	General Address.
Neutral Address	0...65535 Default: 0	

Property/ Parameter	Value or Value range/ Default	Description
Phase A Address	0...65535 Default: 0	Phase A Address.
Phase B Address	0...65535 Default: 0	Phase B Address.
Phase C Address	0...65535 Default: 0	Phase C Address.
Common		
Update Rate	0...60000 Default: 0	Maximum update rate of indication changes between the OPC server and the client in milliseconds. The value 0 means that the server sends all the changes to the client.
Data Specific		
Send as Double Point	True False Default: False	Specifies if a value of an indication signal is sent as a double point value.
Send as Inverse Value	True False Default: False	Specifies if a value of an indication signal is sent as an inverse value.

3.4.5.4. Analogue set point (APC)

Table 3.4.5.4-1 Configurable APC properties

Property/ Parameter	Value or Value range/ Default	Description
Basic		
Common Data Class	APC	Common data class according to IEC 61850.
Addresses		
Control Address	0...65535 Default: 0	Control address.

Property/ Parameter	Value or Value range/ Default	Description
Data Format	1 = WORD 2 = Integer 3 = Long MSW First 4 = Long MSW Last 5 = Float MSW First 6 = Float MSW Last Default: 1	Data format for write value.

3.4.5.5. Binary counter reading (BCR)

Table 3.4.5.5-1 Configurable BCR properties

Property/ Parameter	Value or Value range/ Default	Description
Basic		
Common Data Class	BCR	Common data class according to IEC 61850.
Addresses		
Indication Address	0...65535 Default:0	Indication address.
Data Format	1 = WORD 2 = Integer 3 = Long MSW First 4 = Long MSW Last 5 = Float MSW First 6 = Float MSW Last Default: 1	Data format for counter value.
Common		
Update Rate	0...60000 Default: 0	Maximum update rate of indication changes between the OPC server and the client in milliseconds. The value 0 means that the server sends all the changes to the client.

3.4.5.6. Binary controlled step position information (BSC)

Table 3.4.5.6-1 Configurable BSC properties

Property/ Parameter	Value or Value range/ Default	Description
Basic		
Common Data Class	BSC	Common data class according to IEC 61850.
Addresses		
Control Address	0...65535 Default: 0	Control address.
Indication Address	0...65535 Default: 0	Indication address.
Data Format	1 = WORD 2 = Integer 3 = Long MSW First 4 = Long MSW Last 5 = Float MSW First 6 = Float MSW Last Default: 1	Data format for position value.
Scale	Default: None	Scale used with position information.
Common		
Update Rate	0...60000 Default: 0	Maximum update rate of indication changes between the OPC server and the client in milliseconds. The value 0 means that the server sends all the changes to the client.
Data Specific		
Received as Inverse Control Value	True False Default: False	Specifies if a control value is received as an inverse value.

3.4.5.7. Complex measured value (CMV)

Table 3.4.5.7-1 Configurable CMV properties

Property/ Parameter	Value or Value range/ Default	Description
Basic		
Common Data Class	CMV	Common data class according to IEC 61850.
Addresses		
Address	0...65535 Default: 0	Indication address.
Data Format	1 = WORD 2 = Integer 3 = Long MSW First 4 = Long MSW Last 5 = Float MSW First 6 = Float MSW Last Default: 1	Data format for measurement value.
Scale	Default: None	Scale used with measurement information.
Common		
Update Rate	0...60000 Default: 0	Maximum update rate of indication changes between the OPC server and the client in milliseconds. The value 0 means that the server sends all the changes to the client.

3.4.5.8. Delta (DEL)

Table 3.4.5.8-1 Configurable DEL properties

Property/ Parameter	Value or Value range/ Default	Description
Basic		
Common Data Class	DEL	Common data class according to IEC 61850.
Addresses		
Phase AB Address	0...65535 Default: 0	Phase AB Address.

Property/ Parameter	Value or Value range/ Default	Description
Phase BC Address	0...65535 Default: 0	Phase BC Address.
Phase CA Address	0...65535 Default: 0	Phase CA Address.
Data format Phase AB, BC, CA	1 = WORD 2 = Integer 3 = Long MSW First 4 = Long MSW Last 5 = Float MSW First 6 = Float MSW Last Default: 1	Data format for measurement values.
Scale	Default: None	Scale used with measurement information.
Common		
Update Rate	0...60000 Default: 0	Maximum update rate of indication changes between the OPC server and the client in milliseconds. The value 0 means that the server sends all the changes to the client.

3.4.5.9. Controllable double point (DPC)

Table 3.4.5.9-1 Configurable DPC properties

Property/ Parameter	Value or Value range/ Default	Description
Basic		
Common Data Class	DPC	Common data class according to IEC 61850.
Addresses		
Control Address	0...65535 Default: 0	Address for Control command.
Indication Address	0...65535 Default: 0	Address for Indication.
Common		

Property/ Parameter	Value or Value range/ Default	Description
Update Rate	0...60000 Default: 0	Maximum update rate of indication changes between the OPC server and the client in milliseconds. The value 0 means that the server sends all the changes to the client.
Data Specific		
Received as Inverse Control Value	True False Default: False	Specifies if a control value is received as an inverse value.
Send as Inverse Value	True False Default: False	Specifies if a value of an indication signal is sent as an inverse value.
Send as Single Point	True False Default: False	Specifies if a value of an indication signal is sent as a single point value.

3.4.5.10. Double point status (DPS)

Table 3.4.5.10-1 Configurable DPS properties

Property/ Parameter	Value or Value range/ Default	Description
Basic		
Common Data Class	DPS	Common data class according to IEC 61850.
Addresses		
Indication	0...65535 Default: 0	Indication address.
Common		
Update Rate	0...60000 Default: 0	Maximum update rate of indication changes between the OPC server and the client in milliseconds. The value 0 means that the server sends all the changes to the client.
Data Specific		

Property/ Parameter	Value or Value range/ Default	Description
Send as Inverse Value	True False Default: False	Specifies if a value of an indication signal is sent as an inverse value.
Send as Single Point	True False Default: False	Specifies if a value of an indication signal is sent as a single point value.

3.4.5.11. Controllable integer status (INC)

Table 3.4.5.11-1 Configurable INC properties

Property/ Parameter	Value or Value range/ Default	Description
Basic		
Common Data Class	INC	Common data class according to IEC 61850.
Addresses		
Control Address	0...65535 Default: 0	Control address.
Indication Address	0...65535 Default: 0	Indication address.
Data Format	1 = WORD 2 = Integer 3 = Long MSW First 4 = Long MSW Last 5 = Float MSW First 6 = Float MSW Last Default: 1	Data format for indication value.
Scale	Default: None	Scale used with indication value.
Common		
Update Rate	0...60000 Default: 0	Maximum update rate of indication changes between the OPC server and the client in milliseconds. The value 0 means that the server sends all the changes to the client.

3.4.5.12. Integer status (INS)

Information in the following table applies also to the Internal INS data object.

Table 3.4.5.12-1 Configurable INS properties

Property/ Parameter	Value or Value range/ Default	Description
Basic		
Common Data Class	INS	Common data class according to IEC 61850.
Addresses		
Address	0...65535 Default: 0	Indication address.
Data Format	1 = WORD 2 = Integer 3 = Long MSW First 4 = Long MSW Last 5 = Float MSW First 6 = Float MSW Last Default: 1	Data format.
Scale	Default: None	Scale used with indication value.
Common		
Update Rate	0...60000 Default: 0	Maximum update rate of indication changes between the OPC server and the client in milliseconds. The value 0 means that the server sends all the changes to the client.

3.4.5.13. Integer controlled step position information (ISC)

Table 3.4.5.13-1 Configurable ISC properties

Property/ Parameter	Value or Value range/ Default	Description
Basic		
Common Data Class	ISC	Common data class according to IEC 61850.
Addresses		

Property/ Parameter	Value or Value range/ Default	Description
Control Address	0...65535 Default: 0	Control address.
Indication Address	0...65535 Default: 0	Indication address.
Data Format	1 = WORD 2 = Integer 3 = Long MSW First 4 = Long MSW Last 5 = Float MSW First 6 = Float MSW Last Default: 1	Data format.
Scale	Default: None	Scale used with indication value.
Common		
Update Rate	0...60000 Default: 0	Maximum update rate of indication changes between the OPC server and the client in milliseconds. The value 0 means that the server sends all the changes to the client.

3.4.5.14. Measured value (MV)

Table 3.4.5.14-1 Configurable MV properties

Property/ Parameter	Value or Value range/ Default	Description
Basic		
Common Data Class	MV	Common data class according to IEC 61850.
Addresses		
Address	0...65535 Default: 0	Indication address.

Property/ Parameter	Value or Value range/ Default	Description
Data Format	1 = WORD 2 = Integer 3 = Long MSW First 4 = Long MSW Last 5 = Float MSW First 6 = Float MSW Last Default: 1	Data format.
Scale	Default: None	Scale used with measurement information.
Common		
Update Rate	0...60000 Default: 0	Maximum update rate of indication changes between the OPC server and the client in milliseconds. The value 0 means that the server sends all the changes to the client.

3.4.5.15. Controllable single point (SPC)

Information in the following table applies also to the Internal SPC data object.

Table 3.4.5.15-1 Configurable SPC properties

Property/ Parameter	Value or Value range/ Default	Description
Basic		
Common Data Class	SPC	Common data class according to IEC 61850.
Addresses		
Control Address	0...65535 Default: 0	Control address.
Indication Address	0...65535 Default: 0	Indication address.
Common		
Update Rate	0...60000 Default: 0	Maximum update rate of indication changes between the OPC server and the client in milliseconds. The value 0 means that the server sends all the changes to the client.
Data Specific		

Property/ Parameter	Value or Value range/ Default	Description
Received as Inverse Control Value	True False Default: False	Specifies if a control value is received as an inverse value.
Send as Double Point	True False Default: False	Specifies if a value of an indication signal is sent as a double point value.
Send as Inverse Value	True False Default: False	Specifies if a value of an indication signal is sent as an inverse value.

3.4.5.16. Single point status (SPS)

Information in the following table applies also to the Internal SPS data object.

Table 3.4.5.16-1 Configurable SPS properties

Property/ Parameter	Value or Value range/ Default	Description
Basic		
Common Data Class	SPS	Common data class according to IEC 61850.
Addresses		
Address	0...65535 Default: 0	Address.
Common		
Update Rate	0...60000 Default: 0	Maximum update rate of indication changes between the OPC server and the client in milliseconds. The value 0 means that the server sends all the changes to the client.
Data Specific		
Send as Double Point	True False Default: False	Specifies if a value of an indication signal is sent as a double point value.

Property/ Parameter	Value or Value range/ Default	Description
Send as Inverse Value	True False Default: False	Specifies if a value of an indication signal is sent as an inverse value.

3.4.5.17. WYE

Table 3.4.5.17-1 Configurable WYE properties

Property/ Parameter	Value or Value range/ Default	Description
Basic		
Common Data Class	WYE	Common data class according to IEC 61850.
Subtype		
Subtype	WYE Simple	Subtype of WYE.
Addresses		
Neutral Address	0...65535 Default: 0	Neutral address.
Phase A Address	0...65535 Default: 0	Phase A address.
Phase B Address	0...65535 Default: 0	Phase B address.
Phase C Address	0...65535 Default: 0	Phase C address.
Net Address	0...65535 Default: 0	Net address.
Res Address	0...65535 Default: 0	Res address.

Property/ Parameter	Value or Value range/ Default	Description
Data Format (Phase A, B, C, Net, Res)	1 = WORD 2 = Integer 3 = Long MSW First 4 = Long MSW Last 5 = Float MSW First 6 = Float MSW Last Default: 1	Data format.
Phase Scale	Default: None	Scale used for phase measurement value.
Neutral Scale	Default: None	Scale used for neutral.
Net Scale	Default: None	Scale used for Net.
Res Scale	Default: None	Scale used for Res.
Common		
Update Rate	0...60000 Default: 0	Maximum update rate of indication changes between the OPC server and the client in milliseconds. The value 0 means that the server sends all the changes to the client.

4. Operation

4.1. About this section

This section describes the basic operation procedures you can carry out after the object properties for the Modbus TCP Slave OPC Client have been configured.

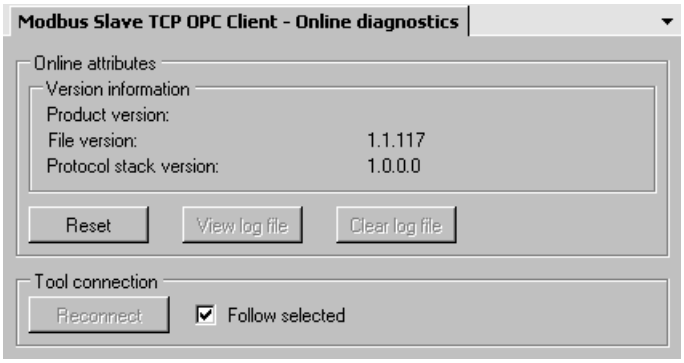
After this you can, for example, monitor and control the condition of connections in the network. This is done by using the Online diagnostics function in SAB600.

4.2. Activating COM600 with new configurations

For information about activating COM600 with new configuration, see COM600 User's Manual.

4.3. Modbus TCP Slave OPC Client diagnostics

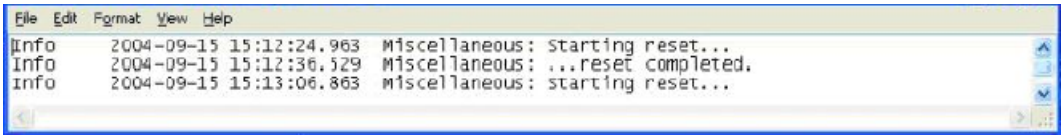
To view version information on Modbus TCP Slave OPC Client or to monitor and control the state of the client, right-click the Modbus TCP Slave OPC Client object and select **Online diagnostics**, see Figure 4.3-1.



Modbus_TCP_Slave_OPC_Client.jpg

Figure 4.3-1 Modbus TCP Slave OPC Client Online diagnostics

In the Online diagnostics box you can:
- reset Modbus TCP Slave OPC Client
- view the event log file, see Figure 4.3-2
- clear the event log file.



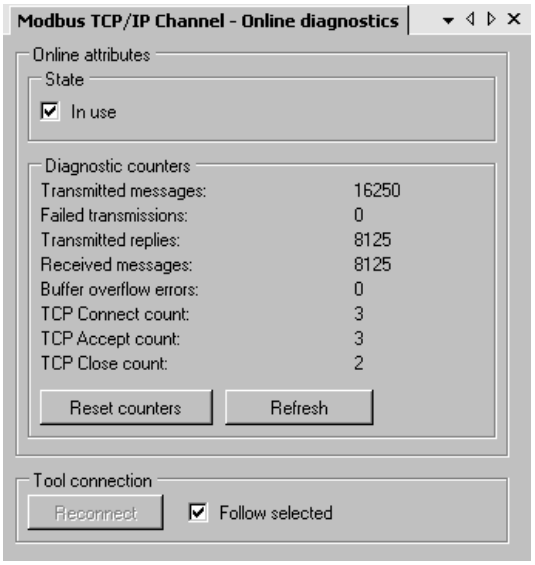
Event_log_file.png

Figure 4.3-2 Event log file

4.4. Monitoring Modbus TCP Channel activity

The Modbus TCP Channel activity can be monitored with the Online diagnostics function. You can also take a channel into use or out of use as described in this section.

To monitor and control Modbus TCP Channel activity:
1. Select the channel you want to monitor in the object tree of SAB600.
2. Right-click the channel.
3. Select **Online diagnostics**.
4. Monitor the channel activity in the **Diagnostic counters** field. The available attributes can be seen in Figure 4.4-1.
5. To reset Diagnostic counters, click **Reset counters**.



Modbus_TCP_Channel_Online_diagnostics.png

Figure 4.4-1 Modbus TCP Channel Online diagnostics

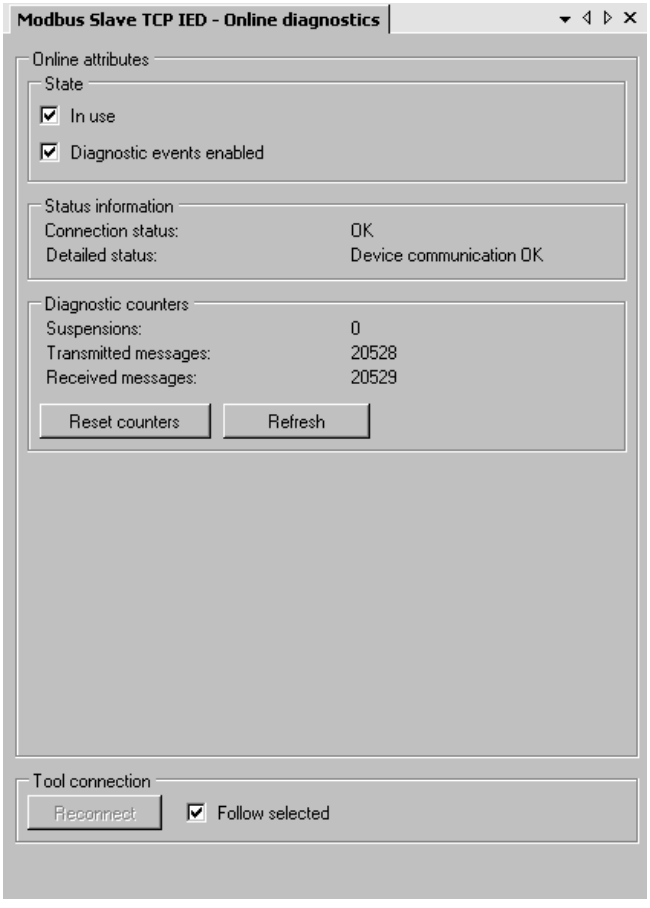
To take a Modbus TCP Channel into use:
1. Select the **In use** check-box. If you clear the check-box, the channel is taken out of use.
2. Update diagnostic counters by clicking **Refresh**.

4.5. Monitoring Modbus TCP IED communication

The Modbus TCP IED communication can be monitored with the Online diagnostics function. You can take a device into use or out of use as described in this section.

To monitor and control Modbus TCP IED communication:
1. Select the device you want to monitor in the object tree of SAB600.
2. Right-click the device.
3. Select **Online diagnostics**.
4. Monitor the device status in the **Status information** field. The **Diagnostic counters** field provides information on the device activity.
5. To reset diagnostic counters, click **Reset counters**.

To take a Modbus TCP IED into use:
1. Select the **In use** check-box. If you clear the check-box, the device is taken out of use.
2. Update diagnostic counters by clicking **Refresh**.



Modbus_TCP_IED_Online_diagnostics.png

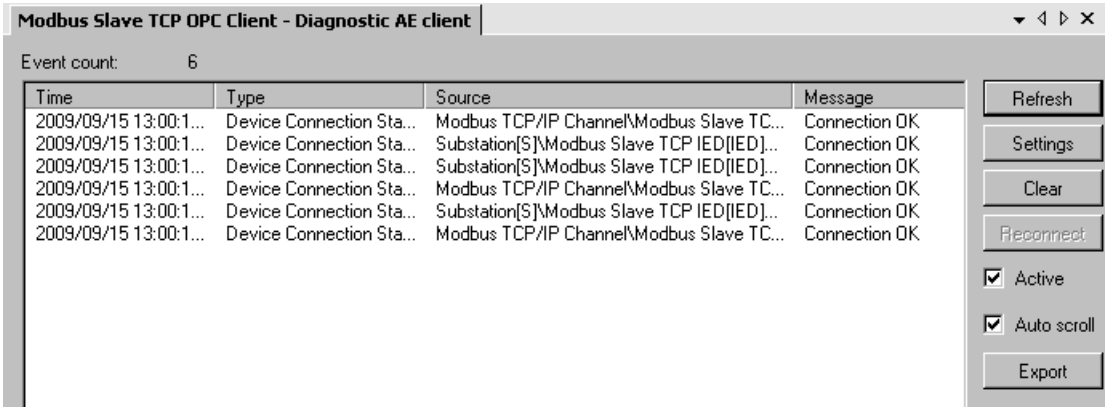
Figure 4.5-1 Modbus TCP IED Online diagnostics

4.6. Viewing events

The Modbus TCP Slave OPC Client has a diagnostic function, which enables monitoring of the flow of process data changes and commands. When the diagnostic function is activated, the Modbus OPC Client Alarm & Event server generates events with information about data changes and commands.

To view the event list:

1. Activate the diagnostics function by selecting the **Diagnostic Events Enabled** check-box, located in the Online diagnostics function of the Modbus TCP IED.
2. Select the Modbus TCP Slave OPC Client object in the object tree of SAB600.
3. Right-click the Modbus TCP Slave OPC Client.
4. Select **Diagnostic AE client** (see Figure 4.6-1).



Modbus_TCP_Slave_OPC_Client_Diagnostics_AE_Client.png

Figure 4.6-1 Modbus TCP Slave OPC Client Diagnostic AE client

5. Technical reference

5.1. About this section

This section provides reference information about the following issues:
- IEC 61850 data modeling
- Attributes
- Status codes.

5.2. IEC 61850 data modeling

5.2.1. General about IEC 61850 data modeling

The following sections describe the relationship between the IEC 61850 data modeling and Modbus TCP Slave OPC Client. There is a table for each data class giving a detailed description about the relationship between the Modbus data and IEC 61850 data object attributes and services. The tables also describe how the data is presented on the OPC Server name space.

The columns in the tables have the following content types:
- **Name** specifies the OPC item name of the attribute/service.
- **Type** specifies the IEC 61850 type of the attribute.
- **Value/ Value range** specifies the allowed values and ranges of the attribute/service.
- **Mandatory/Optional** specifies whether the attribute is considered as mandatory or optional according to the IEC 61850 standard.
- **Modbus information element** specifies the Modbus information element related to the attribute/service.
- **OPC data types** specify the OPC data type used for the OPC item.

5.2.2. Single point status (SPS)

Name	Type	Value/Value range	Mandatory/Optional	Protocol information element	OPC data types
stVal	BOOLEAN	TRUE | FALSE	M		VT_BOOL
q	Quality		M		VT_I4
t	TimeStamp		M		VT_DATE

5.2.3. Double point status (DPS)

Name	Type	Value/ Value range	Mandatory/Optional	Protocol information element	OPC data types
stVal	ENUMERATED	Intermediate-state (0) off (1) on (2) bad-state (3)	M	state (0=OFF, 1=ON)	VT_I4
q	Quality		M	Modbus status	VT_I4
t	TimeStamp		M	<none> | Time of occurence	VT_DATE

5.2.4. Integer status (INS)

Name	Type	Value/ Value range	Mandatory/Optional	Protocol information element	OPC data types
stVal	INTEGER		M	Current value	VT_I4
q	Quality		M	Modbus status	VT_I4
t	TimeStamp		M	<none> | Time of occurrence	VT_DATE

5.2.5. Protection activation information (ACT)

ACT phases information is mapped in the same way as SPS stVal.

Name	Type	Value/ Value range	Mandatory/Optional	Protocol information element	OPC data types
general	BOOLEAN		M	state (0=OFF, 1=ON)	VT_BOOL
phsA	BOOLEAN		O	state (0=OFF, 1=ON)	VT_BOOL
phsB	BOOLEAN		O	state (0=OFF, 1=ON)	VT_BOOL
phsC	BOOLEAN		O	state (0=OFF, 1=ON)	VT_BOOL
neut	BOOLEAN		O	state (0=OFF, 1=ON)	VT_BOOL
q	Quality		M	Modbus status	

Name	Type	Value/ Value range	Mandatory/Optional	Protocol information element	OPC data types
t	TimeStamp		M	<none> | Time of occurence	

5.2.6. Directional protection activation information (ACD)

The ACT-related directional protection activation information is processed like ACT.

Name	Type	Value/ Value range	Mandatory/Optional	Protocol information element	OPC data types
general	BOOLEAN		M	state (0=OFF, 1=ON)	VT_BOOL
phsA	BOOLEAN		O	state (0=OFF, 1=ON)	VT_BOOL
phsB	BOOLEAN		O	state (0=OFF, 1=ON)	VT_BOOL
phsC	BOOLEAN		O	state (0=OFF, 1=ON)	VT_BOOL
neut	BOOLEAN		O	state (0=OFF, 1=ON)	VT_BOOL
q	Quality		M	Modbus status	
t	TimeStamp		M	<none> | Time of occurrence	

5.2.7. Binary counter reading (BCR)

Name	Type	Value/ Value range	Mandatory/Optional	Protocol information element	OPC data types
actVal	INTEGER		M	Value | Frozen value | Current value	VT_I4
q	Quality		M	Modbus status	VT_I4
t	TimeStamp		M	<none> | Time of occurrence	VT_DATE

5.2.8. Measured value (MV)

Name	Type	Value/ Value range	Mandatory/Optional	Protection information element	OPC data types
mag	AnalogueValue		M		VT_R4
q	Quality		M	Modbus status	VT_I4
t	TimeStamp		M	<none> | Time of occurence	VT_DATE

5.2.9. Complex measured value (CMV)

CMV is configured in the same way as MV. The only difference is that, instead of a mag tag, there is a cVal node containing a mag tag in the OPC namespace structure.

Name	Type	Value/ Value range	Mandatory/Optional	Protocol information element	OPC data type
cVal.mag	AnalogueValue		M	CurrentValue	VT_R4
q	Quality		M	Modbus status	VT_I4
t	TimeStamp		M	<none> | Time of occurrence	VT_DATE

5.2.10. WYE

Name	Type	Value/ Value range	Mandatory/Optional	Protocol information element	OPC data types
phsA.cVal.mag	AnalogueValue		M	Phase A Current Value	VT_R4
phsA.q	Quality		M	Modbus status	VT_I4
phsA.t	TimeStamp		M	<none> | Time of occurrence	VT_DATE
phsB.cVal.mag	AnalogueValue		O	Phase B Current Value	VT_R4
phsB.q	Quality		O	Modbus status	VT_I4
phsB.t	TimeStamp		O	<client provided if none>	VT_DATE

Name	Type	Value/ Value range	Mandatory/Optional	Protocol information element	OPC data types
phsC.cVal.mag phsC.q phsC.t	AnalogueValue Quality TimeStamp		O O O	Phase C Current Value Modbus status <none> | Time of occurrence	VT_R4 VT_I4 VT_DATE
neut.cVal.mag neut.q neut.t	AnalogueValue Quality TimeStamp		O O O	Neutral Current Value Modbus status <none> | Time of occurrence	VT_R4 VT_I4 VT_DATE

5.2.11. Delta (DEL)

Name	Type	Value/ Value range	Mandatory/Optional	Protocol information element	OPC data types
phsAB. cVal.mag t phsAB.q phsAB.t	AnalogueValue Quality TimeStamp		M M M	Phase AB Current Value Modbus status <none> | Time of occurrence	VT_R4 VT_I4 VT_DATE
phsBC.cVal.mag q phsBC.q phsBC.t	AnalogueValue Quality TimeStamp		M M M	Phase BC Current Value Modbus status <none> | Time of occurrence	VT_R4 VT_I4 VT_DATE
phsCA.cVal.mag q phsCA.q phsCA.t	AnalogueValue Quality TimeStamp		M M M	Phase CA Current Value Modbus status <none> | Time of occurrence	VT_R4 VT_I4 VT_DATE

5.2.12. Controllable single point (SPC)

Name	Type	Value/ Value range	Mandatory/Optional	Protocol information element	OPC data types
ctlVal	SPI		M	Control Code	VT_BOOL

Name	Type	Value/ Value range	Mandatory/Optional	Protocol information element	OPC data types
stVal		FALSE | TRUE	M	State (0=OFF, 1=ON)	VT_BOOL
q	Quality		M	Modbus status	VT_I4
t	TimeStamp		M	<none> | Time of occurrence	VT_DATE

5.2.13. Controllable double point (DPC)

Name	Type	Value/ Value range	Mandatory/Optional	Protocol information element	OPC data types
ctlOperOn	SPI	FALSE | TRUE	O	Control Code	VT_BOOL
ctlOperOff		FALSE | TRUE	O	Control Code	VT_BOOL
ctlSelOn		FALSE | TRUE	O	Control Code	VT_BOOL
ctlSelOff		FALSE | TRUE	O	Control Code	VT_BOOL
stVal	ENUMERATED	intermediate-state (0) off (1) on (2) bad-state (3)	M	State (0=OFF, 1=ON)	VT_I4
q	Quality		M	Modbus status	VT_I4
t	TimeStamp		M	<none> | Time of occurrence	VT_DATE
ctlCan	BOOLEAN	FALSE | TRUE	O	-	VT_BOOL
stSeld	BOOLEAN	FALSE | TRUE	O	State (0=OFF, 1=ON)	VT_BOOL

5.2.14. Controllable integer status (INC)

Name	Type	Value/ Value range	Mandatory/Optional	Protocol information element	OPC data types
ctlVal	INTEGER		M	Control Value	VT_I4
stVal	INTEGER		M	Current Value	VT_I4
q	Quality		M	Modbus status	VT_I4
t	TimeStamp		M	<none> | Time of occurrence	VT_DATE

5.2.15. Binary controlled step position information (BSC)

Name	Type	Value/ Value range	Mandatory/Optional	Protocol information element	OPC data types
ctlVal	ENUMERATED	stop (0) lower (1) higher (2) reserved (3)	M	Control Value	VT_I1
valWTr	ValWithTrans		M	State	VT_I4
q	Quality		M	Modbus status	VT_I4
t	TimeStamp		M	<none> | Time of occurrence	VT_DATE

5.2.16. Integer controlled step position information (ISC)

Name	Type	Value/ Value range	Mandatory/Optional	Protocol information element	OPC data types
ctlVal	INTEGER	-64 ... 63	M	Control Value	VT_I1
valWTr	ValWithTrans		M	State	VT_I4
q	Quality		M	Modbus status	V_I4
t	TimeStamp		M	<none> | Time of occurrence	V_DATE

5.2.17. Analogue set point (APC)

Name	Type	Value/ Value range	Mandatory/Optional	Protocol information element	OPC data types
spMag	AnalogueValue		M	Control Value	VT_R4

5.3. Attributes

5.3.1. General about attributes

In addition to item tags for process data (indications and commands), the OPC servers and clients also provide some item tags for controlling the devices and retrieving status information from them. These item tags are called attributes.

There are three categories of attributes:
1. Modbus TCP Slave OPC Client attributes
2. Modbus TCP Channel attributes
3. Modbus TCP IED attributes.

These attributes are described in the following subsections.

The server does not automatically update the attributes according to the update rate of the OPC group. Instead, the client must explicitly refresh the group, or request a read of single attributes to retrieve the latest values. This is not necessary for most attributes, since they do not change their value spontaneously. The exceptions are the diagnostic counters and object status attributes. Object status attributes are updated automatically by system messages, so in practice only diagnostic counters are affected by this rule. It was introduced to avoid overloading the communication link between the protocol stack and the OPC server.

5.3.2. Modbus TCP Slave OPC Client attributes

Table 5.3.2-1 Modbus TCP Slave OPC Client attributes

Property / Parameter	Value or Value range/ Default	Description
Protocol Stack Version	Value: Version information	Data type: Text. Access: Read-only. The version information of the Protocol Stack.
Reset		The **Reset** button for resetting the OPC Client.
File Version		File version of the executable OPC Client.
Product Version		Version information of the installed OPC Client.

5.3.3. Modbus TCP Channel attributes

Table 5.3.3-1 Modbus TCP Channel attributes

Property / Parameter	Value or Value range/ Default	Description
In Use	0 = Not in use, the channel communication is stopped 1 = In use Default: 0	Data type: Integer. Access: No limitations. Specifies whether the channel is in use or not. When a channel is not in use, no data can be transmitted on it, and no data is received from it. The channel attributes can be read as usual. Generally, a channel must be taken out of use by setting this attribute to 0 before the channel attributes can be written. When a channel is stopped by setting the In use attribute to 0, all data transmission on the channel ceases. Before that, the protocol stack executes to the end all on-going data transactions. For example, the polling of the station in turn is completed.
Diagnostic Counters (TCP/IP)		
Transmitted Telegrams		The number of transmitted telegrams.
Failed Transmissions		The number of failed transmissions.
Transmitted Commands		The number of transmitted commands.
Transmitted Replies		The number of transmitted replies.
Received Messages		The number of received data messages.
Buffer Overflow Errors		The number of times there has been a buffer overflow.
TCP Connect		Incremented each time a TCP connect request is received.
TCP Accept		Incremented each time a TCP connect request is accepted.
TCP Close		Incremented each time a TCP connection is closed.

5.3.4. Modbus TCP IED attributes

Table 5.3.4-1 Modbus TCP IED attributes

Property / Parameter	Value or Value range/ Default	Description
In Use	0 = Out of use 1 = In use Default: 0	Data type: Integer Access: No limitations The operational status of the device - in use or out of use. Taking the device out of use with this attribute stops all data communication with the device. All operations that would result in a data exchange are disabled. The device itself is not affected by the attribute, only the protocol stack's image of the device. Setting In use to 1 is allowed only if the device address is legal.
Object Status	When written: 1 = Re-transmit system message When read: A status code, for example, 0 = OK (communication works properly) 13801 = Device suspended	Data type: Integer Access: No limitations Indicates the detailed information about the station device status. Writing to the Object Status attribute (Object Status = 1) of a device makes the protocol stack to re-transmit the latest system message caused by the device. Possible "Stopped" and "Suspended" messages cause old marking of OPC items. By reading the Object Status attribute, the status code of the system message can be read.
Device Connection Status	True = Device connection OK False = Device connection suspended	Data type: Boolean Access: Read-only Indicates the status of the device connection.
Diagnostic Counters		
		Data type: Integer Access: No limitations
Suspension Counter		Indicates the number of times the connection has been suspended.

Property / Parameter	Value or Value range/ Default	Description
Transmitted Messages		The number of transmitted messages.
Received Messages		The number of received messages.

Index

A

B

C

D

I

M

O

P

S

V

W

Contact us

ABB Oy
Distribution Automation
P.O. Box 699
FI-65101 VAASA, FINLAND
Tel. +358 10 22 11
Fax. +358 10 224 1094

ABB Inc.
Distribution Automation
655 Century Point
Lake Mary, FL 32746, USA
Tel: +1 407 732 2000
Fax: +1 407 732 2335

www.abb.com/substationautomation

Power and productivity
for a better world™