

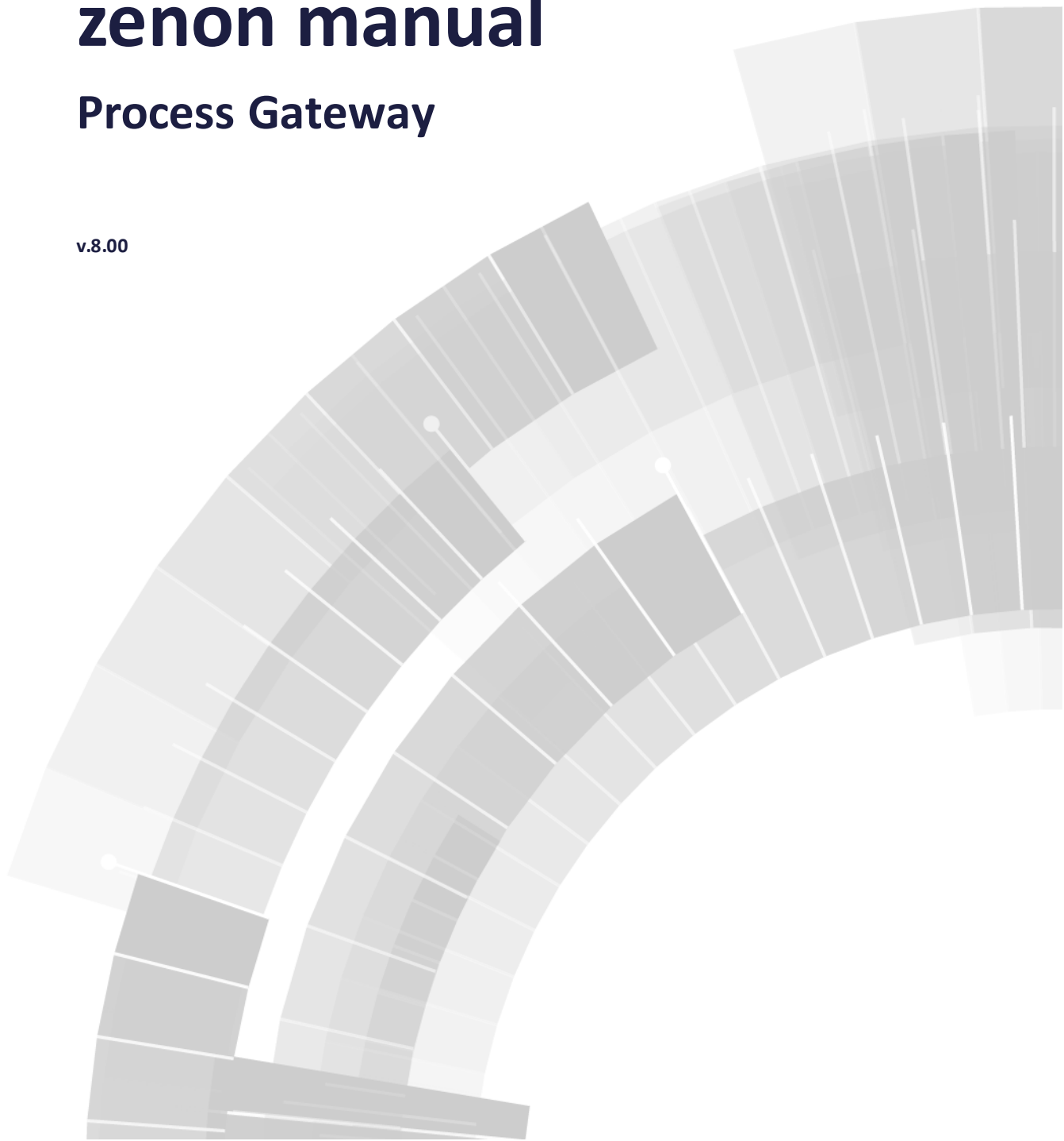


COPADATA
do it your way

zenon manual

Process Gateway

v.8.00





©2018 Ing. Punzenberger COPA-DATA GmbH

All rights reserved.

Distribution and/or reproduction of this document or parts thereof in any form are permitted solely with the written permission of the company COPA-DATA. Technical data is only used for product description and are not guaranteed qualities in the legal sense. Subject to change, technical or otherwise.

Contents

1. Welcome to COPA-DATA help.....	7
2. Process Gateway	7
3. General	8
3.1 Variable assignment using names	9
4. Requirements.....	10
5. Start.....	11
5.1 Start via Startup Tool	13
5.2 Start via zenon function.....	14
5.3 Start with command line.....	16
5.4 Module selection.....	16
5.4.1 First start.....	17
5.4.2 Change existing module selection.....	17
5.4.3 Start with configured module	18
5.5 INI file.....	19
6. DEC	21
6.1 Module configuration.....	21
6.2 Logging	22
7. DNP3_SG.....	22
7.1 Installation.....	23
7.2 Outstation.....	24
7.3 Datalink.....	27
7.3.1 Application Layer	31
7.3.2 Events/Unsolicited	34
7.3.3 File Transfer	38
7.4 Binary Input.....	40
7.5 Double Bit Binary Input.....	48
7.6 Binary Output.....	57
7.7 Running Counter	68

7.8	Frozen Counter	76
7.9	Analog Input.....	84
7.10	Analog Output.....	93
7.11	Octet String	102
7.12	Variable selection dialog.....	108
7.13	Warning dialog.....	109
7.14	Status mapping zenon - DNP3	112
7.15	Select before Operate	114
7.16	Command processing - Command Routing	114
7.17	LATCH_ON and LATCH_OFF	117
8.	DNP3 Slave - AccessDNP3 outdated	119
9.	ICCP-TASE.2.....	120
9.1	Module configuration.....	122
9.1.1	General	124
9.1.2	Server variables.....	131
9.1.3	Client variables.....	135
9.2	Status variables of the connection.....	140
9.3	Show non-present variables in Runtime.....	141
9.4	Direct control (non-SBO) for ICCP control objects	142
9.5	Configuration file: INI entries for ICCP-Tase.2	144
9.6	LOG entries.....	147
9.7	Interoperability.....	151
10.	IEC870 Slave	151
10.1	Module configuration.....	152
10.1.1	General Settings.....	156
10.1.2	Device List.....	159
10.1.3	Device - Configuration	161
10.1.4	Sector - Configuration.....	166
10.1.5	104 settings	188
10.1.6	DPI / DCS mapping	190
10.1.7	Configuration via XML file.....	191
10.2	Sequence of Events (SOE)	201
10.2.1	Warnings	203

10.2.2	Error message	204
10.3	Communication	205
10.3.1	Start.....	205
10.3.2	General interrogation.....	205
10.3.3	End of connection and resets.....	207
10.3.4	Transfer of the value changes	209
10.3.5	Reloading Runtime	209
10.3.6	Data priorities.....	210
10.3.7	Internal type T00 (status variable)	210
10.3.8	Mapping of status bits in message direction.....	215
10.3.9	Commands	217
10.3.10	Read - T102 - request.....	219
10.3.11	Time synchronization - T103 - request	220
10.3.12	Test - T104 and T107 - requests.....	220
10.3.13	File transfer.....	221
10.4	Broadcasts	224
10.5	Error analysis.....	224
10.5.1	Analysis tool.....	225
10.5.2	Error codes.....	230
10.5.3	Check list.....	231
10.6	Interoperability.....	232
11.	MODBUS Slave	250
11.1	Module configuration.....	251
11.2	Holding Registers.....	253
11.3	Coils.....	255
11.4	Address Settings	257
11.5	Hardware address of the MODBUS system	257
11.6	Configuration file: specific entries for AccessMODBUS	258
12.	MS Azure.....	260
12.1	Module configuration.....	261
12.1.1	MS Azure Connection Settings.....	265
12.2	Message formats	269
12.3	Error message.....	270
12.4	Configuration file: specific entries for MS Azure	272

13. OPC UA Server.....	273
13.1 Compare OPC Server and OPC UA Server	274
13.2 Module configuration.....	276
13.2.1 Server	277
13.2.2 Variables.....	283
13.2.3 Archives.....	287
13.3 Start	289
13.4 Connection	290
13.5 Security	290
13.6 Alarms and Conditions	290
13.7 Service Sets.....	293
13.8 Communication	294
13.9 Configuration file: INI entries for OPC UA	297
13.10 Error message.....	300
13.11 Status mapping.....	303
14. SNMP agent	304
14.1 SNMP architecture.....	304
14.2 Management Information Base (MIB).....	305
14.3 MIB structure	305
14.4 Module configuration.....	307
14.5 Configuration file: specific entries for Access SNMP	308
14.6 Configuration of the Windows SNMP agent	308
15. SQL	310
15.1 Module configuration.....	310
15.2 Variables selection.....	311
15.3 Creating tables	312
15.4 Operation in a redundant system	312
15.5 Configuration file: specific entries for AccessSQL.....	313
15.6 Table format.....	314
15.6.1 Table for process image	314
15.6.2 Table for variable IDs	314

1. Welcome to COPA-DATA help

ZENON VIDEO-TUTORIALS

You can find practical examples for project configuration with zenon in our YouTube channel (https://www.copadata.com/tutorial_menu). The tutorials are grouped according to topics and give an initial insight into working with different zenon modules. All tutorials are available in English.

GENERAL HELP

If you cannot find any information you require in this help chapter or can think of anything that you would like added, please send an email to documentation@copadata.com.

PROJECT SUPPORT

You can receive support for any real project you may have from our Support Team, who you can contact via email at support@copadata.com.

LICENSES AND MODULES

If you find that you need other modules or licenses, our staff will be happy to help you. Email sales@copadata.com.

2. Process Gateway

*The **Process Gateway** serves as a coupling to higher-level systems, such as a Control Center. This means that parts of the zenon process image are provided for other applications and updated by these.*

- ▶ Process data from zenon Runtime can be forwarded to superordinate systems.
- ▶ Superordinate systems can write values or commands to zenon Runtime.



Information

The **Process Gateway** user interface is only available in English.

All components used must be licensed.

Note: Both Editor and/or Runtime, which provide the license information, as well as the applications that access it, must run in the same user context. They must be started by the same user. Starting one application as administrator and another as a local user does not work.

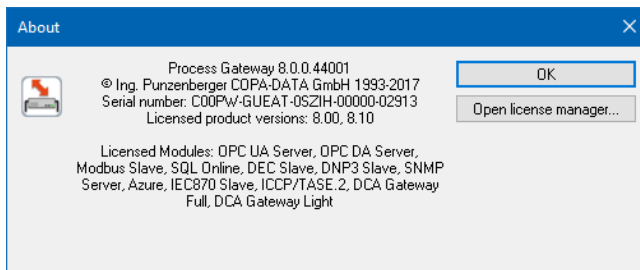
SHOW LICENSE INFORMATION

To view license information in the **Process Gateway**:

1. Open the **Process Gateway**.
2. Click the **About** button.

The **About** dialog is opened.

DIALOG ABOUT



Option	Description
License information	Information on licensed modules including serial number.
OK	Closes the dialog.
Open license manager	Opens the License Manager . Licenses can be activated, returned and managed with this tool. The serial number is needed for this.

If the **Process Gateway** is opened without an existing module configuration, a model must be selected first. After selecting a module, the corresponding start dialog for the **Process Gateway** is opened. In doing so, the **Settings** button is inactive until the **Process Gateway** is licensed.

3. General

The following is applicable for **Process Gateway**:

- ▶ It is thought as an add-on to the Runtime of the control system and only works in combination with it.
That means: If zenon Runtime has not been started, the **Process Gateway** does not start!
- ▶ Several **Process Gateways** can be started on one computer.
You can find further information on this in the Start with command line (on page 16) chapter.
- ▶ The **Process Gateway** can be started on both a standalone computer as well as in the zenon network (server/standby/client). All **Process Gateways** that are started in the network provide the same process image.
- ▶ The module to be started is selected and configured in its own INI file (on page 11) of **Process Gateway**, not in the zenon6.ini file as is otherwise the case for zenon!
The INI file for the **Process Gateway** is called `zenProcGateway.INI` and is in the `%CD_SYSTEM%` folder. With a standard installation, the storage folder is `C:\ProgramData\COPA-DATA\System` by default.

PROJECT SELECTION

If no special project name is defined in the configuration file (`zenProcGateway.ini`), all configurations of the **Process Gateway** are applicable to the first project loaded by Runtime.

In a multi-project system, it is possible to access any desired project in the hierarchy. To do this, enter the project name in the **Project=** entry in the **[GENERAL]** section in the INI file. If this entry is empty, the integration project is applicable.

3.1 Variable assignment using names

In the **Process Gateway**, the respective assignment is carried out using the variable name.

This is important if:

- ▶ Variable names in the project configuration are changed in zenon Editor.
- ▶ Variables in the project configuration in the Editor are deleted.
- ▶ zenon calls up a new start project.

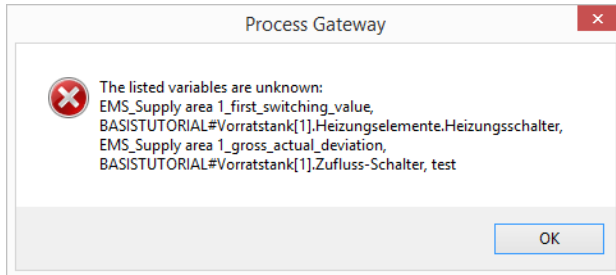
In this case, module-independent error dialogs appear when the **Process Gateway** is started.



Attention

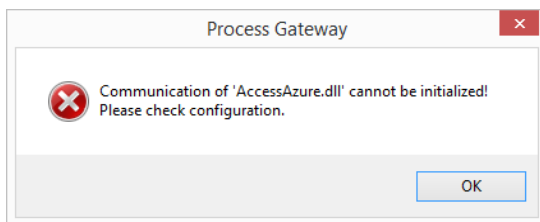
*If a variable assignment is erroneous, the **Process Gateway** is not started! The **Process Gateway** only starts if the incorrect variable assignment has been corrected.*

The first error dialog lists the unknown variables.



- ▶ In this case, amend the variable assignment again.
- ▶ To do this, remove all incorrectly-assigned variables from the respective export area and re-add the corresponding variables.

The dialog that follows contains module-dependent error messages.



LOG ENTRY

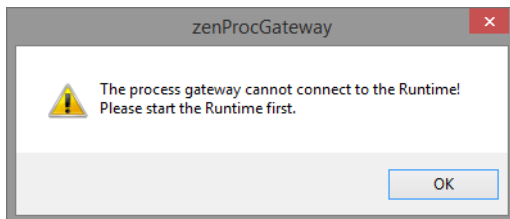
The following LOG entry is created in the event of incorrect variable assignment:

LOG entry	Description
Failed to update value for variable %s.	No COM connection to zenon could be established, or the variable is not (or no longer) available in zenon or has been renamed.

4. Requirements

On the computer on which the **Process Gateway** should run, the Runtime with the corresponding release version and a loaded project that corresponds to the configuration has to be started before it is started. If Runtime is stopped, the **Process Gateway** is also stopped automatically.

If zenon Runtime is not loaded when **Process Gateway** is started, a corresponding warning dialog is shown:



Information

The **Process Gateway** uses the COM interface to access zenon data. The COM interface is used by VBA and is only available if **zenon6.ini** is present in the file under **[VBA]** the entry **EVENT=1**.

5. Start

The **Process Gateway** consists of the **zenProcGateway.exe** file. This is copied to the zenon program folder during installation. However, the file can also be used if it is saved in another folder.

Attention: The module (Access*.dll) to be used with the **Process Gateway** has to be in the same folder as **zenProcGateway.exe**.

Exception: **SNMP** (on page 308).

The **zenProcGateway.ini** configuration file is in the %CD_SYSTEM% system folder.

In this INI file, the **DLL=** entry (in the **[GENERAL]** section) determines the module with which the **Process Gateway** starts.

Several **Process Gateways** can be started on one computer. The prerequisite for this is the use of different INI files. You can find further information on this in the Start with command line (on page 16) chapter.

PROCESS GATEWAY IN ZENON NETWORK

Process Gateway can be started on both a standalone computer as well as in the zenon network (server/standby/client). All **Process Gateways** that are started in the network provide the same process image.

In redundant networks, it is recommended that **Process Gateway** is started on a (dedicated) client. With redundancy switching, a network client automatically connects to the respective primary server and continues to provide the data to higher-level systems.

PROCESS GATEWAY - START

Process Gateway can be started:

- ▶ via Startup Tool (on page 13)
- ▶ via zenon function (on page 14)

In both start methods, the **Process Gateway** can be started with parameters in the command line interface (on page 16). Different INI files of the **Process Gateway** can thus be determined.



Attention

Process Gateway can only be started if zenon Runtime is running. Starting without Runtime running is not possible. A corresponding warning dialog is shown in this case.

CONFIGURATION

The configuration consists of two parts.

- ▶ Module selection (on page 16)
In the first stage of configuration, select the module that communicates with the **Process Gateway**.
You can find further information on this in the module selection chapter.
- ▶ Module-specific configuration
Depending on the selected module, connection parameters and variable assignments can be configured.
You can find details on this from the module configuration chapter in the respective section in this handbook.

All configurations of the **Process Gateway** are saved in an INI file. Please note that not all entries of the corresponding INI file can be configured via a corresponding user interface.

The module is determined with the **DLL=** entry (in the **[GENERAL]** section).

Possible entries:

Module	Entry
DEC	DLL=AccessDEC.dll
SQL	DLL=AccessSQL.dll
MODBUS	DLL=AccessMODBUS.dll
DNP3	DLL=AccessDNP3.dll
IEC870SI	DLL=AccessIEC870SI.dll
SNMP	DLL=AccessSNMP.dll
OPCUA	DLL=AccessOPCUA.dll
ICCP/TASE.2	DLL=AccessICCP.dll
MS Azure	DLL=AccessAzure.dll
DNP3_SG	DLL=AccessDNP3_SG.dll

SECURITY CONFIGURATION

The **Process Gateway** can be executed hidden to protect it from unauthorized access. If the **Process Gateway** is started in the command line with the parameter `/hide` (or `-hide`), it starts invisible and cannot be configured or stopped. The **Process Gateway** closes automatically when the zenon Runtime is closed.

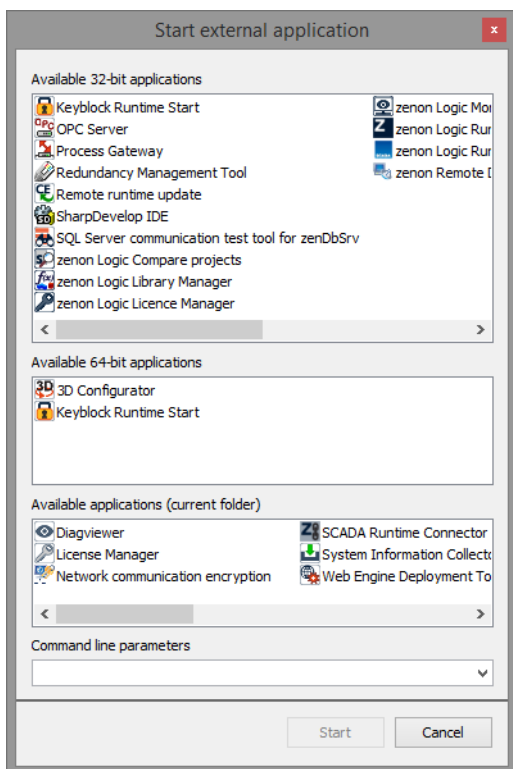
5.1 Start via Startup Tool

The **Process Gateway** can be started via the zenon **Startup Tool**.

To do this, carry out the following steps:

1. Open the zenon **Startup Tool**.
2. Click **Tools**.

The dialog for configuring the zenon Tools is opened.



3. Select **Process Gateway**.
4. Click the Start button to start the **Process Gateway** with the configuration from the current `zenProcGateway.INI` file.

Or:

Enter the name of the EXE file and the name of the INI file as a start parameter into the **Command line parameters**:

zenProcGateway.EXE /ini:"MyConfig.INI"

You can find further information on this in the Start with command line (on page 16) chapter.

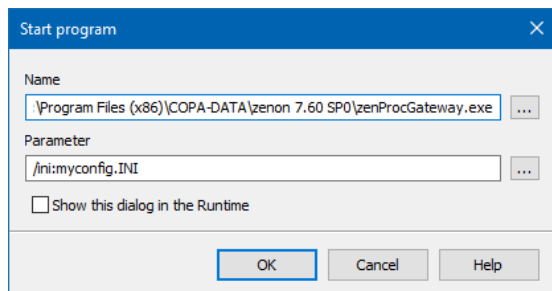
Note: The parameters alone are not sufficient. The **zenProcGateway.EXE** call must be entered.

5. Click **Start**.

5.2 Start via zenon function

The **Process Gateway** can - like any other Windows application - be started with the **Start program** function.

ENGINEERING



Steps to create the function:

1. Create a new function:
In the toolbar or in the context menu of the Functions node, select **New function**.
The dialog to select a function is opened.
2. Navigate to the node **Windows**.
3. Select the **Start program** function.
The dialog to select the program and enter start parameters is opened.
4. Select the desired program.
 - a) Click on the button ...
The file selection dialog is opened.
 - b) Go to the zenon Installation directory:
Program Files (x86) - 64-bit operating system
Program Files - 32-bit operating system
 - c) Select the file `zenProcGateway.exe`
 - d) Confirm the selection by clicking on the **Open** button
5. Optional: Configure the parameters that are executed when the function is executed.
You can find further information on this in the Start with command line (on page 16) chapter.
If this parameter is empty, the **Process Gateway** starts with the `zenProcGateway.INI` file.
6. Close the dialog by clicking on the **OK** button.
7. Name the function in the **Name** property.



Hint

If this function is linked in the `AUTOSTART` script, **Process Gateway** automatically starts with Runtime.*

5.3 Start with command line

Several **Process Gateways** can be started on one computer. To do this, a separate INI file must be present for each **Process Gateway** to be started. The INI file must be in the system folder (%CD_SYSTEM%).

Examples of use:

- ▶ Simultaneous start of several **Process Gateways** with different modules
- ▶ Start several instances of the **Process Gateway** with the same module (not applicable for all modules).
Requirement: separate communication parameters for each instance, for example no conflicts in IP ports.

In this case, the **zenProcGateway.EXE** must be started with the command line parameter **/ini:"[file]"**.

Note: the name of the INI file can also be configured for starting with zenon function (on page 14).

Syntax of command line:

- ▶ In the Startup Tool:
`zenProcGateway.EXE /ini:"[FileName].INI"`
- ▶ In the zenon **Start program** function
Parameter input field: `/ini:"[FileName].INI"`

`[FileName].ini` must be replaced by the correct name of the INI file.

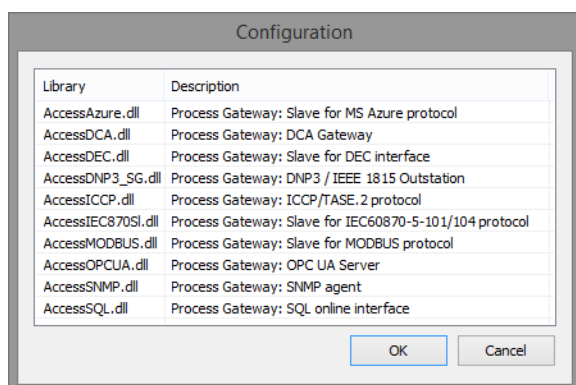
5.4 Module selection

The following possibilities are available for the selection of a module for the **Process Gateway**:

- ▶ **First start**
When the **Process Gateway** is started for the first time, a dialog to select the module is shown. This selection dialog is also offered when **Process Gateway** is started if there is no INI file or if no module is configured in the selected INI file.
The `zenProcGateway.INI` file is used by default.
- ▶ **Module already configured**
If a module has already been configured for **Process Gateway**, the Process Gateway starts with the communication dialog of the configured module.
When **Process Gateway** starts, the communication already starts if all necessary connection parameters for the module have been configured and are valid.

5.4.1 First start

The module must be configured when it is started for the first time.



To start **Process Gateway** with a selectable module:

1. Start **Process Gateway** with an INI file that does not exist (there must be no other module configured in the INI file).
The **Process Gateway** uses the file `zenProcGateway.INI` by default. You can define another file in the command line (`/ini:"[FileName].INI"`).
You can find more information in the **Start with command line** (on page 16).
2. The dialog to select the module is opened.
3. Select the desired module.
4. Confirm your selection by clicking on the **OK** button.
The **Process Gateway** then creates an INI file.
5. The **Process Gateway** starts with the communication dialog (on page 18) of the selected module.

5.4.2 Change existing module selection

If the Process Gateway starts with a module that has already been configured for a different communication protocol, without offering the dialog to select a module, then:

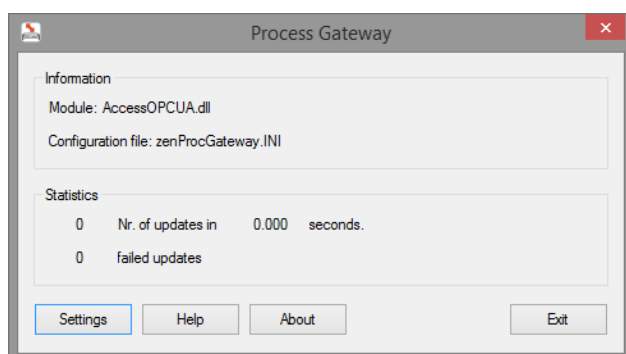
1. Close the **Process Gateway**.
2. Edit the existing `zenProcGateway.INI` file.
 - a) To do this, go to the `%CD_SYSTEM%` directory.
 - b) Open the file in any desired text editor and add comments for the module in the file; or rename the INI file.

3. Restart the **Process Gateway**:

If the INI file has been removed, the dialog to select a module (on page 17) is offered again when **Process Gateway** starts.

5.4.3 Start with configured module

Start the **Process Gateway** with a module that has already been configured; a dialog with static communication data is opened.



INFO

Minimum information about the configured module.

Parameter	Description
Module:	Configured and started module of Process Gateway .
Config File:	Name of the zenProcGateway.INI configuration file of Process Gateway .

STATISTICS

Information on communication with the counterparty:

Parameter	Description
Nr. of updates in xx seconds	Number of successful communication attempts incl. total duration.
failed updates	Number of failed communication attempts.
Exit	Closes the Process Gateway .
Settings	Opens the module's configuration dialog.
Help	Opens online help.



Attention

*If the application is started as **invisible**, there is also no configuration dialog displayed. If the configuration is invalid or not present, the **Process Gateway** is automatically closed.*

5.5 INI file

Process Gateway saves the module selection in the INI file. You can find this `zenProcGateway.INI` file in the following path: `%CD_SYSTEM%`. The configuration is saved for each module as an XML file. This configuration file is also saved in the `%CD_SYSTEM%` folder.

The module-specific configurations between the modules are very different. Please note the detailed documentation in the module-dependent entries in this manual.

Due to the system, only ANSI and Unicode are supported for reading the INI files.



Attention

UTF-8 format is not supported!

You should therefore always save your INI files as a text file in ANSI or Unicode format.

MODULE-INDEPENDENT CONTENT

[GENERAL]

Entry	Description
[GENERAL]	General setting for Process Gateway, regardless of the modules selected.
DLL=	<p>Selection of the DLL file that is to be used for Process Gateway. The selected DLL file determines the module selection</p> <ul style="list-style-type: none"> ▶ AccessAzure.dll ▶ AccessDEC.dll ▶ AccessDNP3_SG.dll ▶ AccessICCP.dll ▶ AccessIEC870SI.dll ▶ AccessMODBUS.dll ▶ AccessOPCUA.dll ▶ AccessSNMP.dll ▶ AccessSQL.dll

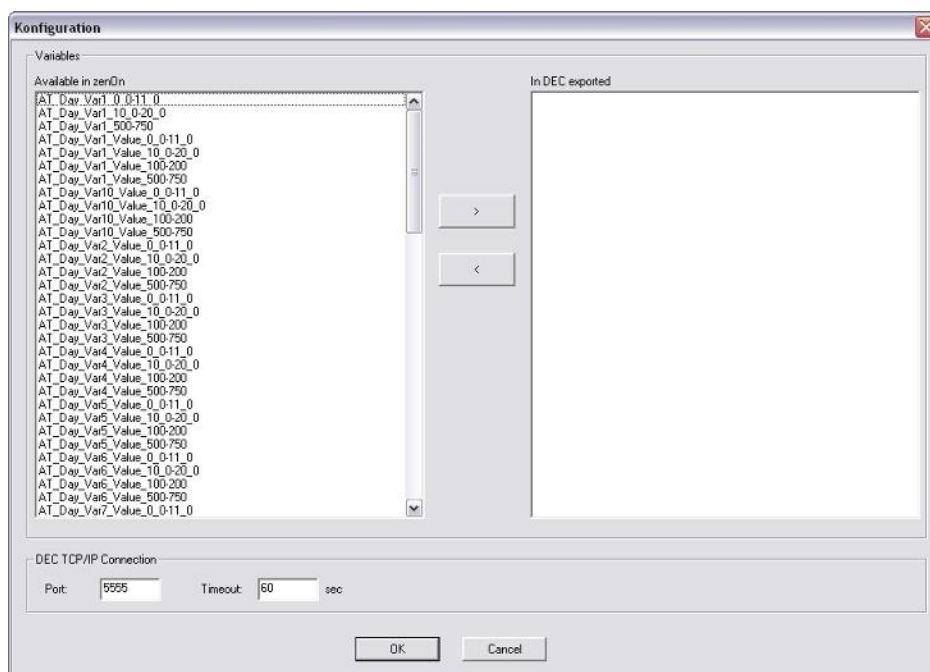
[DATABASE]

Entry	Description
PROJECT=	<p>Name of the zenon project with which the Process Gateway communicates with Runtime.</p> <p>In a multi-project system, it is possible to access any desired project in the hierarchy. If this entry is empty, the integration project is applicable.</p> <p>Default: Empty (project currently loaded in Runtime)</p>

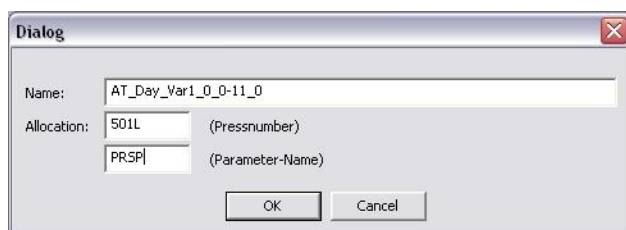
6. DEC

6.1 Module configuration

The following dialog appears after the **Settings** button is clicked:



Here the variables that should be connected can be selected. In the list on the left-hand side all variables which are available in zenon are displayed. These can be selected and moved to the DEC export list with the '>' button. With the button '<' they can be removed again. If a variable is moved to the export list, a dialog for the name assignment of the zenon variable to the DEC variable opens:



The assignment is defined with a four digit press number and the four digit parameter name. Entered characters are automatically changed to upper case. The variables selected in this way then are continuously synchronized in zenon or in the connected DEC system.

In the lower part of the configuration dialog the TCP/IP connection to the DEC system can be configured.

The port number indicates the port where the Process Gateway waits for incoming connections.

The timeout is the time without response, after that the connection is closed.

6.2 Logging

All telegrams received by the DEC system and the responds from the Process Gateway can be logged.

The Diagnosis Viewer is used for logging; you can find further information in the Diagnosis Viewer chapter.

7. DNP3_SG

The **AccessDNP3_SG** module for the **Process Gateway** is based on a DNP3 stack developed by COPA-DATA.

- ▶ The **AccessDNP3_SG** module is compliant with subset level 1, 2 and 3 and supports file transfer.
- ▶ The configuration is not compatible with the existing **AccessDNP3 Process Gateway** of zenon versions before version 8.00.
- ▶ The previous **AccessDNP3** module in the **Process Gateway** can no longer be selected for new configurations.
- ▶ Existing configurations with the previous **AccessDNP3** module continue to be able to run.
- ▶ The **AccessDNP3** module will no longer be supplied in a future version. The old module will be completely replaced by the **AccessDNP3_SG** module.

An XML device profile is supplied with the new module. This document also contains a description of the abilities of the **AccessDNP3_SG** module.

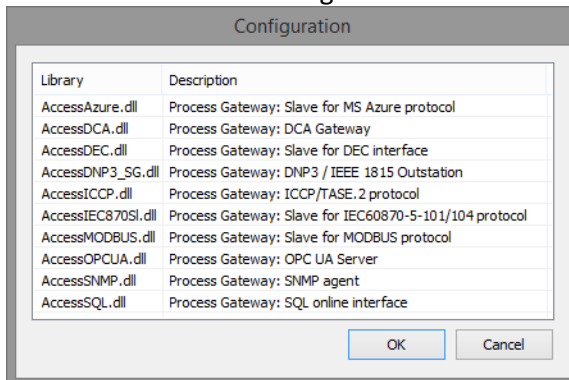
Installation path:

C:\ProgramData\COPA-DATA\zenon8.00\CommunicationProfiles\Dnp3\ProcessGateway

7.1 Installation

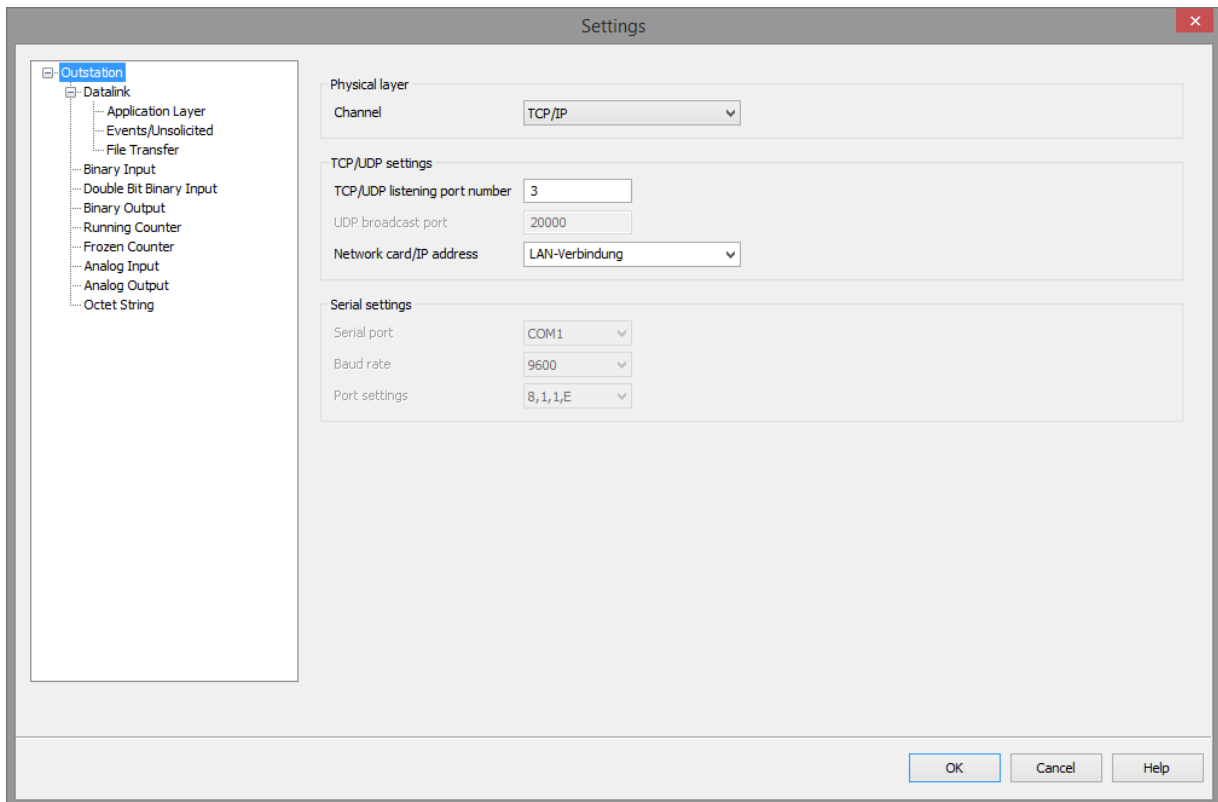
You have the following possibilities for installing the DNP3_SG Process Gateway:

1. The first time Process Gateway is started. No module has yet been configured:
The module must be configured the first time it is started.



- a) Start **Process Gateway** with an empty file or an INI file that does not exist (there must be no other module configured in the INI file).
The **Process Gateway** uses the file `zenProcGateway.INI` by default. You can define another file in the command line (`/ini:"[FileName].INI"`).
You can find more information in the **Start with command line** (on page 16).
 - b) The **Configuration** dialog to select the module is opened.
 - c) Select **AccessDNP3_SG.dll** as a module.
 - d) Confirm your selection by clicking on the **OK** button.
The **Process Gateway** then creates an INI file.
 - e) **Process Gateway** starts with the Process Gateway communication dialog.
 - f) Click on the **Settings** button to configure the **DNP3_SG** module.
2. Start with a **Process Gateway** module that has already been configured.
If **Process Gateway** starts with a module that has already been configured for a different communication protocol, without offering the dialog to select a module, then carry out the following steps:
 - a) Close the **Process Gateway**.
 - b) Edit the existing `zenProcGateway.INI` file.
 - c) To do this, go to the `%CD_SYSTEM%` directory.
 - d) Open the file in any desired text editor and add comments for the module in the file; or rename the INI file.
 - e) Restart the **Process Gateway**:
If the INI file has been removed, the dialog to select a module (on page 17) is offered again when **Process Gateway** starts.

7.2 Outstation



In this dialog, you configure the connection parameters to the outstation.

PHYSICAL LAYER

Parameter	Description
Channel	<p>Type of connection to the outstation. Select from drop-down list:</p> <ul style="list-style-type: none"> ▶ Serial Serial connection to the outstation. ▶ TCP/IP Connection to the outstation via TCP/IP protocol. ▶ UDP Connection to the outstation via UDP protocol. ▶ TCP with UDP broadcast Connection to the outstation via TCP/IP protocol with UDP broadcast. <p>Default: TCP/IP</p>

TCP/UDP SETTINGS

Parameter	Description
TCP/UDP listening port number	<p>Port number for the TCP or UDP communication with the outstation.</p> <p>Default: 20000</p>
UDP Broadcast port	<p>Port number for the receipt of UDP broadcasts.</p> <p>Default: 20000</p> <p>Note: only active if, in the Channel option, the TCP with UDP broadcast value has been configured.</p>
Network card/IP address	<p>Network interface for communication with the outstation. Select from drop-down list.</p> <p>The outstation receives network telegrams at the configured interface. If no interface is selected, the outstation receives the telegrams at all interfaces.</p> <p>Default: empty</p> <p>Note: Not active if Serial is configured for the Channel option.</p>

SERIAL SETTINGS

The options for this group are only active if the `Serial` connection type has been configured for the `Channel` option.

Parameter	Description
Serial port	Selection of the serial interface. Selection from a drop-down list. Default: COM1
Baud rate	Selection of the Baud rate for serial communication. Select from drop-down list. Default: 9600
Port settings	Number of data bits, start bits and stop bits, as well as the parity for serial communication. Select from drop-down list: <ul style="list-style-type: none"> ▶ 8, 1, 1, N ▶ 8, 1, 1, E ▶ 8, 1, 2, E ▶ 8, 1, 2, N Default: 8, 1, 1, E

7.3 Datalink

Settings

Outstation

- Datalink
- Application Layer
- Events/Unsolicited
- File Transfer
- Binary Input
- Double Bit Binary Input
- Binary Output
- Running Counter
- Frozen Counter
- Analog Input
- Analog Output
- Octet String

Datalink

Outstation DNP Address: 1

Master DNP Address: 100

Link status request interval: 0 s

Data Link Confirmations: ☒

Data Link retries: 0

Data Link timeout: 2000 ms

Self address response: ☒

Validate master address: ☒

Channel

Master IP Address: 127.0.0.1

Use specific master UDP port: ☐

Master UDP Port: 20000

OK Cancel Help

Parameter	Description
Outstation DNP Address	DNP3 address of the outstation. Default: 1
Self address response	<p>Action of the outstations in response to a <code>Request</code>.</p> <ul style="list-style-type: none"> ▶ Active: The outstation responds with its own DNP3 address if a <code>Master Request</code> is received with the "0xFFFC" reserved address. ▶ Inactive: The outstation only responds to <code>Requests</code> from the configured master address. <p>Default: <code>Inactive</code></p> <p>Note: It is recommended that this option is only activated temporarily to allow a master to detect the outstation.</p>
Master DNP Address	<p>The DNP3 address of the master that is to communicate to the outstation.</p> <p>Default: 100</p> <p>Note: the Process Gateway currently only supports one master per instance.</p>
Validate master address	<p>Validation of the master address</p> <p>Note: This option is always active. The outstation only responds to <code>Master requests</code> that come from the configured Master DNP Address. An exception is the broadcast addresses, if <code>TCP with UDP Broadcast</code> has been configured in the Channel option in the Outstation node and the <code>Broadcasts allowed</code> option has been activated in the Application Layer node.</p> <p>Default: <code>active</code></p>
Link status request interval	<p>Intervals in seconds in which a Datalink Layer <code>REQUEST_LINK_STATUS</code> is sent from the outstation to the master, if there is no further data traffic.</p> <p>This is, for configurations whereby no cyclical communication takes place with the master, to monitor whether the master is still available.</p> <p>Default: 0 With the value 0, the outstation does not send any link status requests.</p> <p>Note: There is a TCP Keep Alive for TCP connections. This</p>

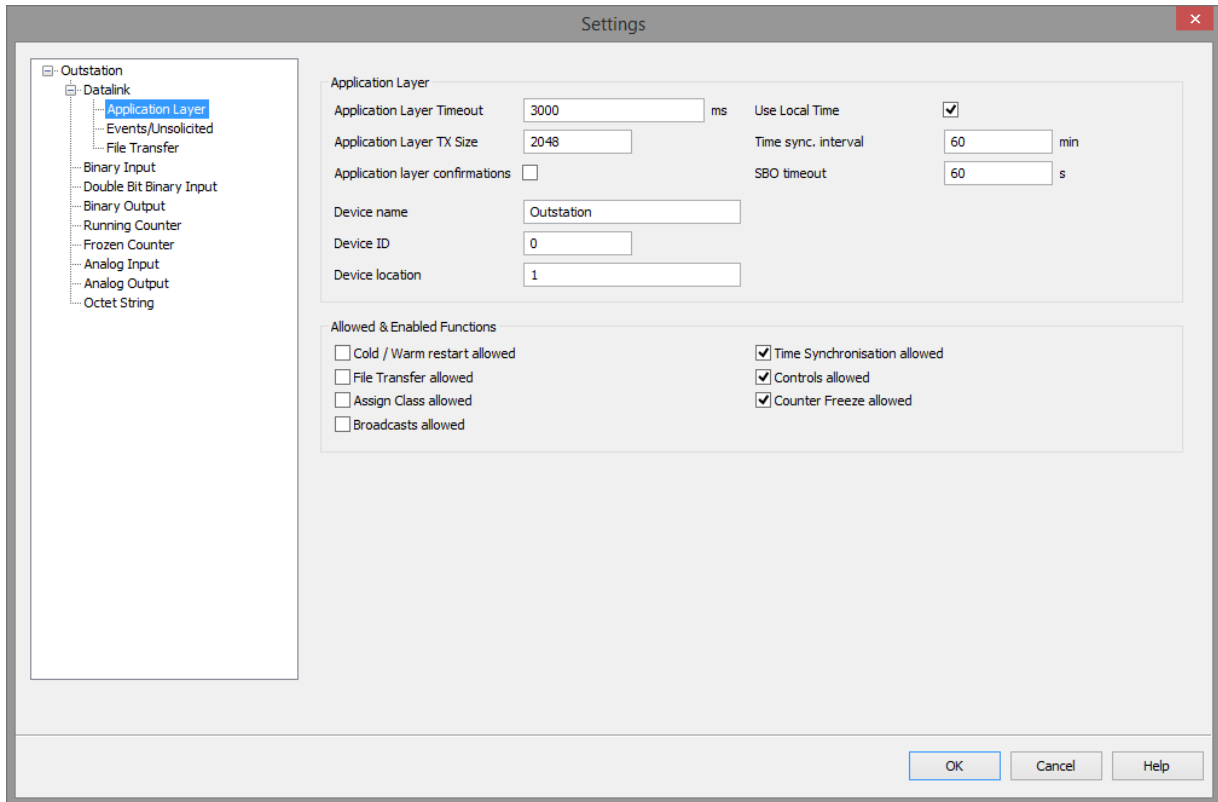
	must not be confused with the Link status Requests.
Data Link Confirmations	<ul style="list-style-type: none"> ▶ Active: The outstation requests a Datalink Layer confirmation from the master for each Datalink Frame. ▶ Inactive: The outstation does not request confirmation for Datalink Layer Frames. <p>Default: Inactive</p> <p>Note: This option should only be activated if it requires a low-quality connection. For TCP/IP connections, it is strongly recommended that this option is not activated.</p>
Data Link retries	<p>Number of recurrences for a Datalink Layer Frame if a confirmation has been requested from the master and this confirmation has not been received within the time configured in the Datalink Layer Timeout option.</p> <p>Default: 0</p>
Data Link Timeout	<p>Timeout for the receipt of a Datalink Layer confirmation from the master to a Datalink Layer Request requested by an outstation for which a Datalink Layer Confirmation has been requested.</p> <p>Default: 2000</p>

CHANNEL

Parameter	Description
Master IP Address	<p>IP address of the remote DNP3 master.</p> <p>Only network telegrams from the IP address configured here are accepted.</p> <p>Exception: Requests At the UDP broadcast address if the Channel option has been configured with TCP/IP with UDP Broadcasts in the Outstation node and the Broadcasts allowed option is active in the Application Layer node.</p> <p>Default: 127.0.0.1</p> <p>Note: Not active if Serial is configured for the Channel option.</p>
Use specific master UDP port	<ul style="list-style-type: none"> ▶ Active: The outstation responds to the UDP port

	<p>configured here.</p> <ul style="list-style-type: none"> ► Inactive: The outstation responds to the UDP port from which the <code>Master Request</code> has been sent. <p>Only available if the Channel has been configured with <code>UDP</code> in the Outstation node.</p>
Master UDP Port	<p>This port is used for the optional <code>unsolicited null response message</code>.</p> <p>Alternative UDP port number for the response to the master if the Use specific master UDP port option has been activated.</p> <p>Default: 20000</p> <p>Note: Only available if the Channel has been configured with <code>UDP</code> in the Outstation node.</p>

7.3.1 Application Layer



The screenshot shows a 'Settings' dialog box with a tree view on the left and configuration fields on the right. The tree view shows a hierarchy: Outstation > Datalink > Application Layer (selected). The configuration fields are organized into two sections: 'Application Layer' and 'Allowed & Enabled Functions'.

Application Layer

Application Layer Timeout	3000	ms	Use Local Time	<input checked="" type="checkbox"/>
Application Layer TX Size	2048		Time sync. interval	60 min
Application layer confirmations	<input type="checkbox"/>		SBO timeout	60 s
Device name	Outstation			
Device ID	0			
Device location	1			

Allowed & Enabled Functions

<input type="checkbox"/> Cold / Warm restart allowed	<input checked="" type="checkbox"/> Time Synchronisation allowed
<input type="checkbox"/> File Transfer allowed	<input checked="" type="checkbox"/> Controls allowed
<input type="checkbox"/> Assign Class allowed	<input checked="" type="checkbox"/> Counter Freeze allowed
<input type="checkbox"/> Broadcasts allowed	

At the bottom right of the dialog are three buttons: OK, Cancel, and Help.

Parameter	Description
Application Layer Timeout	<p>Time that the outstation waits for a response from the master if Responses demand a confirmation.</p> <p>Default: 3000 ms</p>
Use Local Time	<p>Time format used.</p> <p>This option is applicable both for time synchronization of the outstation as well as for the Event Objects that contain a time stamp.</p> <ul style="list-style-type: none"> ▶ Active: The time format uses the local time / time zone of the computer. This option can be set if the master also considers the time as local time. ▶ Inactive: The time format is UTC, corresponding to the IEEE Std 1815tm standard. <p>Default: active</p>
Application Layer TX Size	<p>Maximum size of an Application Layer response. The value can be amended according to the ability of the master and the bandwidth of the connection</p> <p>Default: 2048</p>
Time sync. interval	<p>Time period after which the outstation sets the Internal Indication Bit IIN1.4 (NEED_TIME) in the Response again.</p> <p>The value can be set to 0 if no time synchronization is required by the master.</p> <p>Example: The outstation master already synchronizes its time from another source.</p> <p>Default: 60 min</p>
Application layer confirmations	<p>Application Confirmation for Response</p> <ul style="list-style-type: none"> ▶ Active: The outstation requests an Application Confirmation from the master for each Response. ▶ Inactive: The outstation only requests an Application Confirmation from the master for the following Responses.

	<ul style="list-style-type: none"> - Response contains events - unsolicited Responses - multi-fragment responses (From the first to the second-to-last fragment. The last fragment does not demand a Confirmation) <p>Default: Inactive</p> <p>Note: Only activate this option in exceptional cases, for example if the connection quality and bandwidth requires this (low bandwidth).</p>
SBO timeout	<p>Time that the master waits for an Operate Request after a Select Request has been received by the master.</p> <p>Default: 60 s</p> <p>You can find further information about the SBO in the Select before Operate (on page 114) chapter.</p>
Device name	<p>Description for the outstation.</p> <p>This option can be read and written by the master using the User-assigned device name (g0v247) Device Attribute</p> <p>Default: Outstation</p>
Device ID	<p>Identification for the outstation.</p> <p>This option can be read and written by the master using the User-assigned ID code/number (g0v246) Device Attribute</p> <p>Default: 0</p>
Device location	<p>Site description for the outstation.</p> <p>This option can be read and written by the master using the User-assigned location name (g0v245) Device Attribute</p> <p>Default: 1</p>

ALLOWED & ENABLED FUNCTIONS

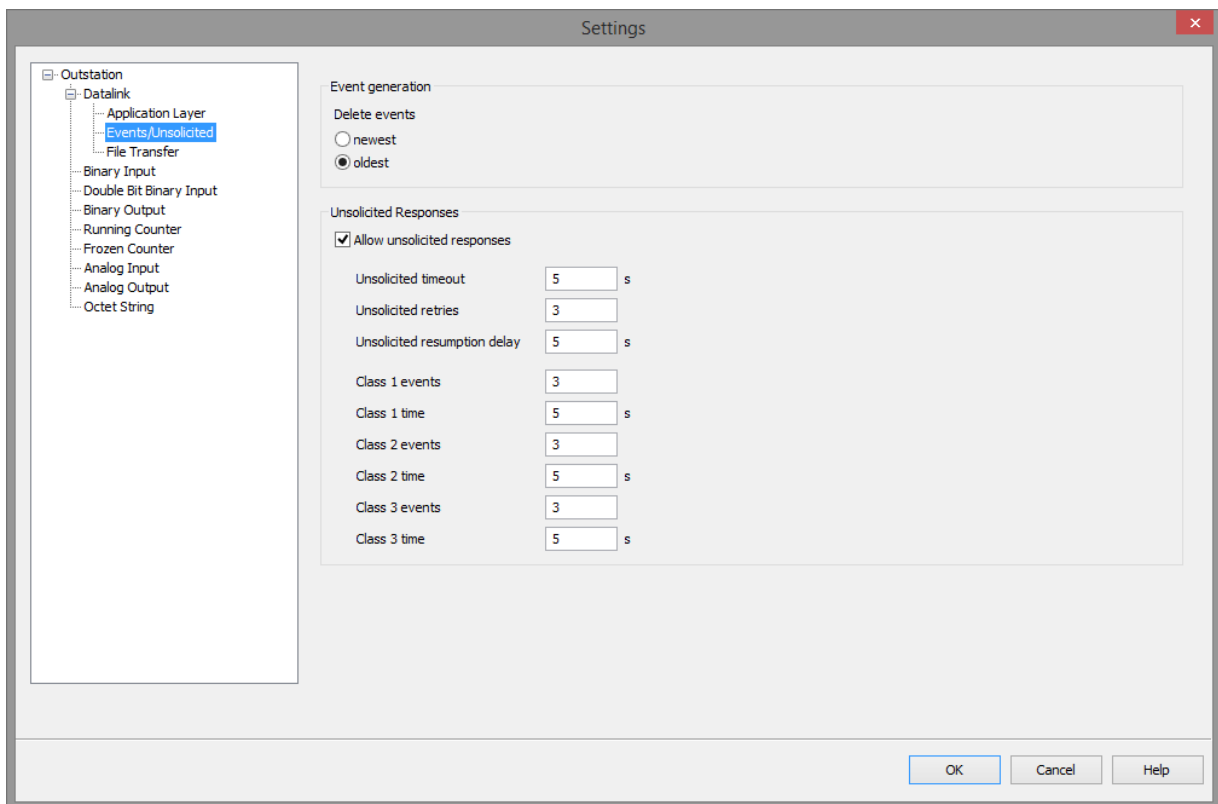
Functions that are offered by the outstation for the connection between the master and outstation.

If an option is not active, the outstation responds to the master with a **Null Response** with the Internal Indication Bit IIN2.0 (NO_FUNC_CODE_SUPPORT).

Exception: there is generally no response to Broadcasts requests.

- ▶ **Cold/Warm restart allowed**
Default: *Inactive*
- ▶ **File Transfer allowed**
Default: *Inactive*
- ▶ **Assign Class allowed**
Default: *Inactive*
- ▶ **Broadcasts allowed**
Default: *Inactive*
- ▶ **Time Synchronisation allowed**
Default: *active*
- ▶ **Controls allowed**
Default: *active*
- ▶ **Counter Freeze allowed**
Default: *active*

7.3.2 Events/Unsolicited



EVENT GENERATION

Parameter	Description
Delete events	<p>The events are deleted in the event of an overflow of the Outstation event buffer:</p> <ul style="list-style-type: none"> ▶ newest: Most recent (=youngest) events. ▶ oldest: Oldest events <p>Default: newest</p>

UNSOLICITED RESPONSES

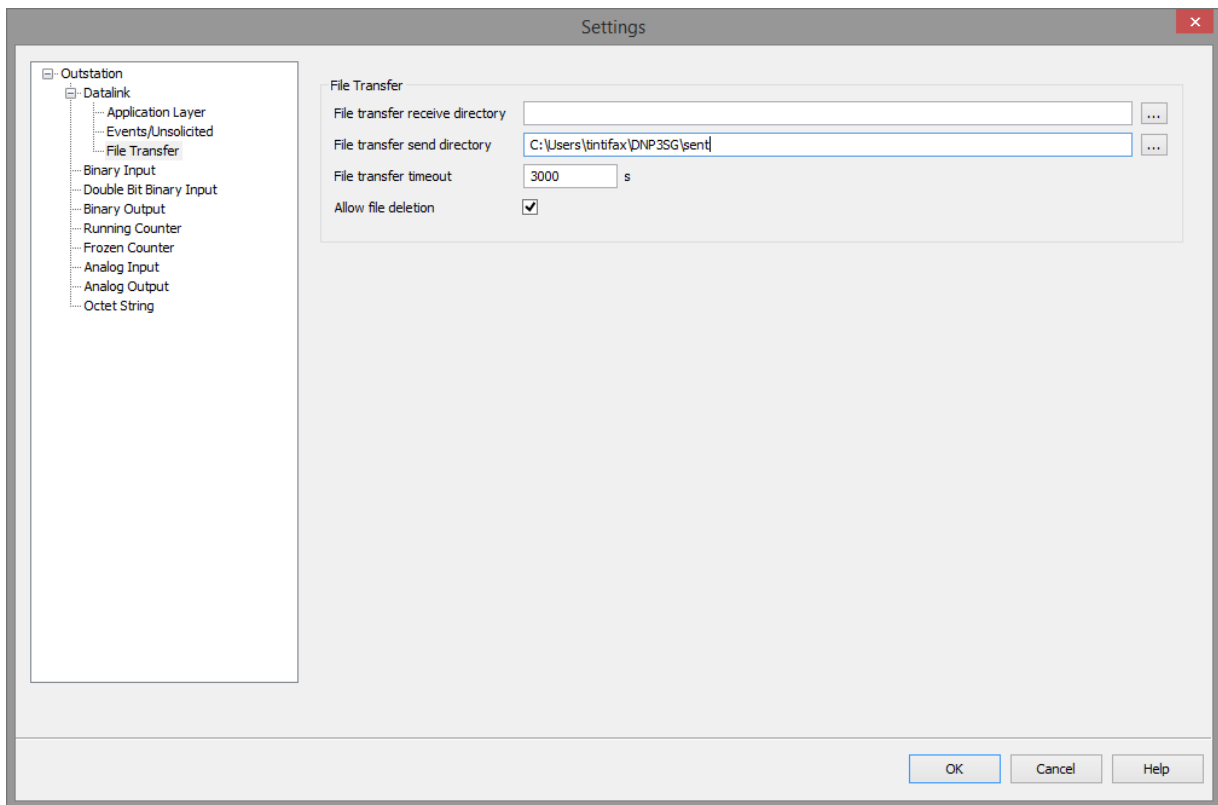
The options in this area are only available if the **Allow unsolicited responses** option has been activated.

Parameter	Description
Allow unsolicited responses	<p>Check box to activate the use of Unsolicited responses.</p> <ul style="list-style-type: none"> ▶ Active: Unsolicited responses are permitted and can be activated by the master. ▶ Inactive: Unsolicited responses are not permitted and cannot be activated by the master. Attention: No unsolicited responses are sent to the master. Even if these have been requested by the master. <p>Default: Inactive</p> <p>Note: Match the configuration of this option to the configuration of the master.</p>

Unsolicited timeout	<p>Time that is waited for an Application Confirmation by the master for a previous unsolicited Response from the outstation.</p> <p>Input range: 1 to 60 s</p> <p>Default: 5 s</p>
Unsolicited retries	<p>Number of recurrences for an Unsolicited Response (sent by the outstation) after the Unsolicited timeout time has expired without an Application Confirmation being received by the master.</p> <p>Input range: 0 – 65535 (0 = an infinite amount of attempts)</p> <p>Default: 3</p> <p>Example: If an Unsolicited Response is sent to the master, the master should send an Application Confirmation. If this confirmation from the master remains outstanding, a repeat is sent as many times as configured in Number of retries. A new Unsolicited Response is then created.</p>
Unsolicited resumption delay	<p>Number of seconds that the outstation waits until it continues sending Unsolicited Responses again, if, after expiry of the Unsolicited timeout time and the sending of the repetitions, no Application Confirmation has been received by the master.</p> <p>Default: 15 s</p>
Class 1 events	<p>Maximum number of events for the event class that can be buffered before an Unsolicited Response is sent to the master.</p> <p>Input range: 0 – 255</p> <p>Default: 3</p>
Class 1 time	<p>Maximum delay of the event class that is waited until an Unsolicited Response is sent to the master after an event has occurred.</p> <p>Input range: 0 – 65535 seconds (0 = no delay, i.e. for each event)</p> <p>Default: 5</p> <p>Note: If new events are continually generated, it may be the case that no Unsolicited Response is sent to the master if the value is greater than 0.</p>

Class 2 events	<p>Maximum number of events for the event class that can be buffered before an <code>Unsolicited Response</code> is sent to the master.</p> <p>Input range: 0 – 255 Default: 3</p>
Class 2 time	<p>Maximum delay of the event class that is waited until an <code>Unsolicited Response</code> is sent to the master after an event has occurred.</p> <p>Input range: 0 – 65535 seconds (0 = no delay, i.e. for each event) Default: 5</p> <p>Note: If new events are continually generated, it may be the case that no <code>Unsolicited Response</code> is sent to the master if the value is greater than 0.</p>
Class 3 events	<p>Maximum number of events for the event class that can be buffered before an <code>Unsolicited Response</code> is sent to the master.</p> <p>Input range: 0 – 255 Default: 3</p>
Class 3 time	<p>Maximum delay of the event class that is waited until an <code>Unsolicited Response</code> is sent to the master after an event has occurred.</p> <p>Input range: 0 – 65535 seconds (0 = no delay, i.e. for each event) Default: 5</p> <p>Note: If new events are continually generated, it may be the case that no <code>Unsolicited Response</code> is sent to the master if the value is greater than 0.</p>

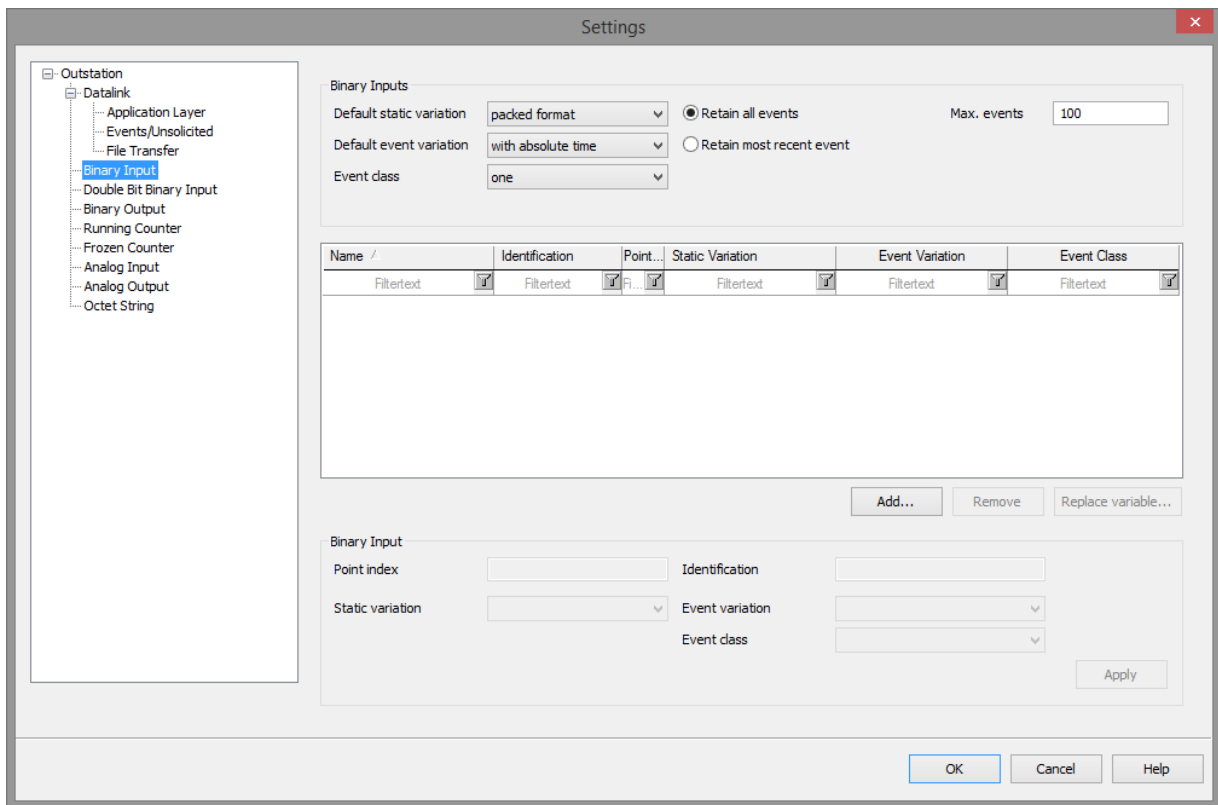
7.3.3 File Transfer



In this dialog, you configure the parameters for file transfer from and to the outstation.
A master can access subdirectories in the save location but not higher-level directories (\..\..\)

Parameter	Description
File transfer receive directory	<p>Save location for the files sent to the outstation by the master.</p> <p>Click the ... button and the dialog opens to select a save location.</p> <p>Note: If the save location is configured differently to the File Transfer send directory option, the master can neither read nor delete the files that it writes itself.</p>
File transfer send directory	<p>Save location for files that can be read by the master.</p> <p>Click the ... button and the dialog opens to select a save location.</p>
File transfer timeout	<p>Time in seconds until the outstation closes the File Handle provided this File Handle has not already been closed by the master.</p> <p>Default: 3000 s</p> <p>Note: The outstation only supports a single file handle.</p>
Allow file deletion	<p>Deletion by the master</p> <ul style="list-style-type: none"> ▶ Active: Files can be deleted by the master. ▶ Inactive: The master cannot delete any files. <p>Default: active</p>

7.4 Binary Input



You configure the DNP3 **Binary Input** points in this node.

BINARY INPUTS

Parameter	Description
Default static variation	<p>This option determines the default static variation for this Object Group for newly-added variables. A different static variation can be individually configured for each variable.</p> <p>Note: Select a static variation that is supported by the master. Not every master supports all static variations.</p> <p>Select from drop-down list:</p> <ul style="list-style-type: none"> ▶ packed format Object Group 1 - Variation 1 ▶ with flags Object Group 1 - Variation 2 <p>Default: packed format</p>
Event retention	<p>Behavior when forwarding value changes.</p> <p>Selection from option field.</p> <ul style="list-style-type: none"> ▶ Retain all events Each time a value, status or time stamp of a variable is changed, a new event is created and saved in the event buffer. The master gets all changes. ▶ Retain most recent event The event buffer buffer only contains one event for the last change of value, status or time stamp. The master only gets the last change and, under certain circumstances, cannot distinguish whether or how often a value has changed in the meantime. <p>Default: Retain all events</p>

Max. events	<p>Maximum number of events that are saved in the event buffer for this Object Group for all defined variables. If <code>Retain all events</code> is configured, this number should be at least as large as the number of configured variables.</p> <p>Default: 100</p> <p>Note: Only active if <code>Retain all events</code> is active.</p>
Default event variation	<p>This option determines the default event variation for this Object Group for newly-added variables. A different event variation can be individually configured for each variable.</p> <p>Note: Select a event variation that is supported by the master. Not every master supports all event variations.</p> <p>Select from drop-down list:</p> <ul style="list-style-type: none"> ▶ without time Object Group 2 - Variation 1 ▶ with absolute time Object Group 2 - Variation 2 ▶ with relative time Object Group 2 - Variation 3 <p>Default: with absolute time</p>
Default event class	<p>This option determines the default event class for this Object Group for newly-added variables. A different event class can be individually configured for each variable.</p> <p>Select from drop-down list:</p> <ul style="list-style-type: none"> ▶ none Variables are not assigned to an event class and do not generate events. No static objects are included for the variables in a response to a class 0 poll from the master. A master can only read the current value of the variables by means of a read request for the respective static object group. A master can assign the variables by means of an assign class request, not another event class. ▶ zero Variables are assigned to the event class 0, but do

	<p>not generate any events. In a response to a class 0 poll from the master, the static objects are included for the variables. A master can also read the variables by means of a read request for the respective static object group. A master can assign the variables by means of an assign class request, the event class 1, 2 or 3, as a result of which the variables also generate events from this point in time.</p> <ul style="list-style-type: none"> ▶ one Variables are assigned to the event class 1, and generate events. In a response to a class 0 poll from the master, the static objects are included for the variables. A master can also read the current value of the variables by means of a read request for the respective static object group. In a response to a class 1 poll from the master, any event objects that may be present are included for the events of the variables. A master can assign the variables by means of an assign class request, the event class 0, 2 or 3. If the variable is assigned by the master of event class 0, no more events are generated. ▶ two Variables are assigned to the event class 2, and generate events. In a response to a class 0 poll from the master, the static objects are included for the variables. A master can also read the current value of the variables by means of a read request for the respective static object group. In a response to a class 2 poll from the master, any event objects that may be present are included for the events of the variables. A master can assign the variables by means of an assign class request, the event class 0, 1 or 3. If the variable is assigned by the master of event class 0, no more events are generated. ▶ three Variables are assigned to the event class 3, and generate events. In a response to a class 0 poll from the master, the static objects are included for the variables. A master can also read the current value of the variables by means of a read request for the respective static object group. In a response to a class 3 poll from the master, any event objects that may be present are included for the events of the variables. A master can assign the variables by means of an assign class request, the event class 0, 1 or 2. If the variable is assigned by the master of event class 0, no more events are generated. <p>Default: one</p>
--	---

Variables taken from the zenon project are listed with their current configuration in this list.

- ▶ The list can be sorted
 - Click for the sorting on the column heading.
 - The sorting sequence is visualized with an arrow symbol next to the column heading:
 Arrow upwards: ascending sorting
 Arrow downwards: descending sorting
 - Another click on the column heading reverses the sorting order.
- ▶ The list can be filtered
 To filter the list:
 - Enter the desired filter term in the input field below the heading. The default description of an empty field is `filter text` (shown in gray font).

Parameter	Description
Name	Variable name. The project name is placed in front of the name: <code>ProjectName#VariableName</code>
Identification	Identification of the variable.
Point Index	Current configuration of the DNP3 point index of the variable in this Object Group.
Static Variation	Current configuration of the static variation for the point of the variable in this Object Group.
Event Variation	Current configuration of the event variation for the point of the variable in this Object Group. Note: This parameter is not available in the Octet String node.
Event Class	Current configuration of the event class for the point of the variable in this Object Group.
Control Model	Current configuration of the Control Model . Note: This parameter is only available in the Binary Output node.
Command Routing	Current behavior of the Binary Output point with select and operate requests from the master. Note: This parameter is only available in the Binary Output node.

Supported data types: `BOOL`, `USINT`

NAVIGATION BAR

Parameter	Description
Add...	<p>Opens the variable selection dialog (on page 108) to apply the zenon variables in the list.</p> <p>Note: Not active in the Frozen Counter node.</p>
Remove	<p>Removes selected variables from the list.</p> <p>A confirmation dialog is shown before a variable is deleted.</p> <p>Only active if one or more variables in the list are selected.</p> <p>Note: Not active in the Frozen Counter node.</p>
Replace variable...	<p>Opens the variable selection dialog (on page 108) to replace a variable selected in the list with another zenon variable.</p> <p>Only active if precisely one variable in the list is selected.</p>

BINARY INPUT

You configure the parameters for a selected variable in the list in this area. Editing by means of multiple selection is not possible.

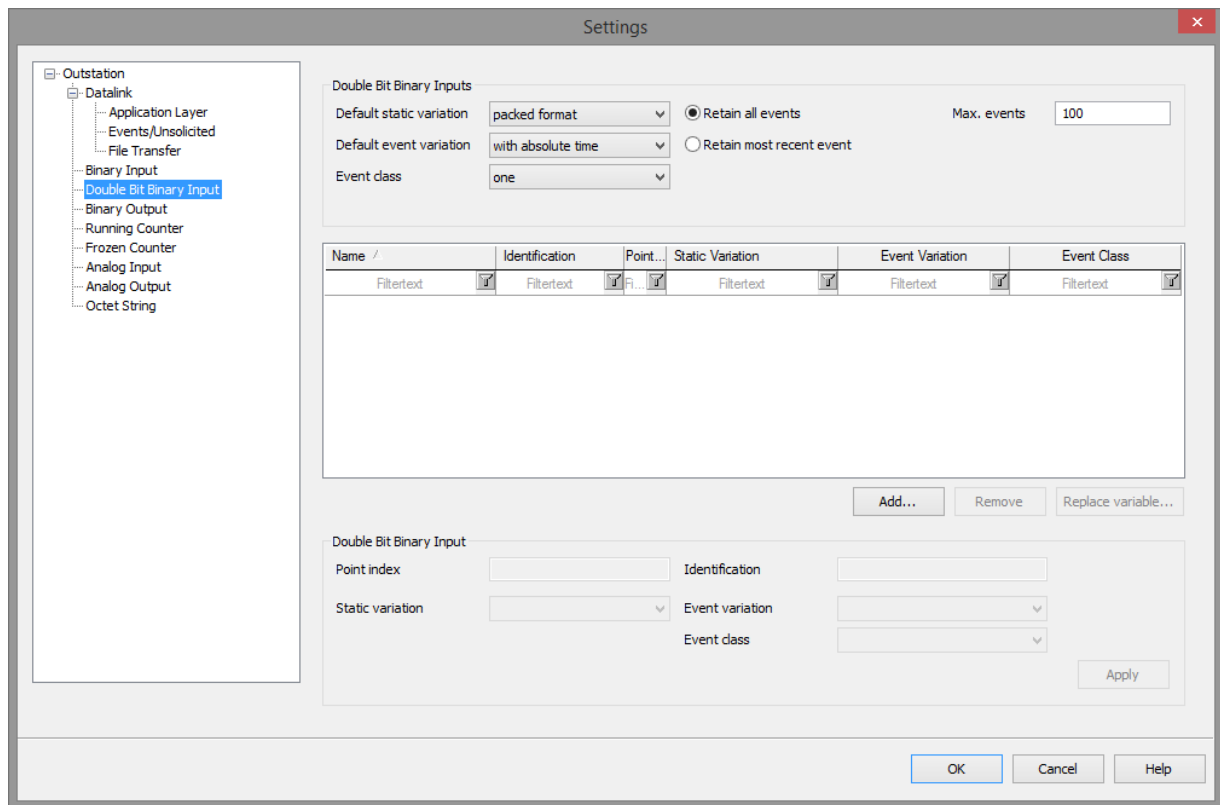
Parameter	Description
Point index	<p>This option determines the DNP3 point index in the Object Group for the selected variable. If variables are added, the point index is automatically set for the newly-added variables and numbered consecutively starting with the highest point index already present.</p> <p>A point index can only be issued once within an Object Group. The entry is validated. If a point index has already been issued, this is shown with a warning dialog.</p> <p>Note: It is recommended that you issue the point index starting with 0 without gaps. The outstation supports the maximum point index. If a higher point index is used, it should be ensured that the master also supports qualifier codes with 2 octets or 4 octets accordingly.</p>
Identification	<p>The identification is primarily taken from the variable from zenon Runtime. The identification can also be amended locally with this option.</p>
Static variation	<p>This option determines the static variation for the selected variable for the point in this Object Group. With the <code>default</code> value, the general static variation defined for this Object Group is used.</p> <p>Note: Select a static variation that is supported by the master. Not every master supports all static variations.</p> <p>Select from drop-down list:</p> <ul style="list-style-type: none"> ▶ <code>packed format</code> Object Group 1 - Variation 1 ▶ <code>with flags</code> Object Group 1 - Variation 2 <p>Default: <code>default</code></p>

Event variation	<p>This option determines the event variation for the selected variable for the point in this Object Group. With the <code>default</code> value, the general event variation defined for this Object Group is used.</p> <p>Note: Select a event variation that is supported by the master. Not every master supports all event variations.</p> <p>Select from drop-down list:</p> <ul style="list-style-type: none"> ▶ without time Object Group 2 - Variation 1 ▶ with absolute time Object Group 2 - Variation 2 ▶ with relative time Object Group 2 - Variation 3 <p>Default: <code>default</code></p>
Event class	<p>This option determines the event class for the selected variable for the point in this Object Group. With the <code>default</code> value, the general event class defined for this Object Group is used.</p> <p>Selection: see description for Default event class option.</p> <p>Default: <code>default</code></p>
Apply	Applies the configuration for the selected variable(s).

CLOSE DIALOG

Options	Description
OK	Applies settings and closes the dialog.
Cancel	Discards all changes and closes the dialog.
Help	Opens online help.

7.5 Double Bit Binary Input



You configure the DNP3 Double bit binary Input points in this node.

Parameter	Description
Default static variation	<p>This option determines the default static variation for this Object Group for newly-added variables. A different static variation can be individually configured for each variable.</p> <p>Note: Select a static variation that is supported by the master. Not every master supports all static variations.</p> <p>Select from drop-down list:</p> <ul style="list-style-type: none"> ▶ packed format Object Group 3 - Variation 1 ▶ with flags Object Group 3 - Variation 2 <p>Default: packed format</p>
Event retention	<p>Behavior when forwarding value changes.</p> <p>Selection from option field.</p> <ul style="list-style-type: none"> ▶ Retain all events Each time a value, status or time stamp of a variable is changed, a new event is created and saved in the event buffer. The master gets all changes. ▶ Retain most recent event The event buffer buffer only contains one event for the last change of value, status or time stamp. The master only gets the last change and, under certain circumstances, cannot distinguish whether or how often a value has changed in the meantime. <p>Default: Retain all events</p>
Max. events	<p>Maximum number of events that are saved in the event buffer for this Object Group for all defined variables. If Retain all events is configured, this number should be at least as large as the number of configured variables.</p> <p>Default: 100</p> <p>Note: Only active if Retain all events is active.</p>
Default event variation	<p>This option determines the default event variation for this Object Group for newly-added variables. A different event variation can be individually configured for each variable.</p>

Note: Select a event variation that is supported by the master. Not every master supports all event variations.

Select from drop-down list:

- ▶ without time
Object Group 4 - Variation 1
- ▶ with absolute time
Object Group 4 - Variation 2
- ▶ with relative time
Object Group 4 - Variation 3

Default: with absolute time

Event class	<p>This option determines the default event class for this Object Group for newly-added variables. A different event class can be individually configured for each variable.</p> <p>Select from drop-down list:</p> <ul style="list-style-type: none"> ▶ none <p>Variables are not assigned to an event class and do not generate events. No static objects are included for the variables in a response to a class 0 poll from the master. A master can only read the current value of the variables by means of a read request for the respective static object group. A master can assign the variables by means of an assign class request, not another event class.</p> ▶ zero <p>Variables are assigned to the event class 0, but do not generate any events. In a response to a class 0 poll from the master, the static objects are included for the variables. A master can also read the variables by means of a read request for the respective static object group. A master can assign the variables by means of an assign class request, the event class 1, 2 or 3, as a result of which the variables also generate events from this point in time.</p> ▶ one <p>Variables are assigned to the event class 1, and generate events. In a response to a class 0 poll from the master, the static objects are included for the variables. A master can also read the current value of the variables by means of a read request for the respective static object group. In a response to a class 1 poll from the master, any event objects that may be present are included for the events of the variables. A master can assign the variables by means of an assign class request, the event class 0, 2 or 3. If the variable is assigned by the master of event class 0, no more events are generated.</p> ▶ two <p>Variables are assigned to the event class 2, and generate events. In a response to a class 0 poll from the master, the static objects are included for the variables. A master can also read the current value of the variables by means of a read request for the respective static object group. In a response to a class 2 poll from the master, any event objects that may be present are included for the events of the variables. A master can assign the variables by</p>
--------------------	---

means of an assign class request, the event class 0, 1 or 3. If the variable is assigned by the master of event class 0, no more events are generated.

► `three`

Variables are assigned to the event class 3, and generate events. In a response to a class 0 poll from the master, the static objects are included for the variables. A master can also read the current value of the variables by means of a read request for the respective static object group. In a response to a class 3 poll from the master, any event objects that may be present are included for the events of the variables. A master can assign the variables by means of an assign class request, the event class 0, 1 or 2. If the variable is assigned by the master of event class 0, no more events are generated.

► Default: `one`

Variables taken from the zenon project are listed with their current configuration in this list.

- ▶ The list can be sorted
 - Click for the sorting on the column heading.
 - The sorting sequence is visualized with an arrow symbol next to the column heading:
 Arrow upwards: ascending sorting
 Arrow downwards: descending sorting
 - Another click on the column heading reverses the sorting order.
- ▶ The list can be filtered
 To filter the list:
 - Enter the desired filter term in the input field below the heading. The default description of an empty field is `filter text` (shown in gray font).

Parameter	Description
Name	Variable name. The project name is placed in front of the name: <code>ProjectName#VariableName</code>
Identification	Identification of the variable.
Point Index	Current configuration of the DNP3 point index of the variable in this Object Group.
Static Variation	Current configuration of the static variation for the point of the variable in this Object Group.
Event Variation	Current configuration of the event variation for the point of the variable in this Object Group. Note: This parameter is not available in the Octet String node.
Event Class	Current configuration of the event class for the point of the variable in this Object Group.
Control Model	Current configuration of the Control Model . Note: This parameter is only available in the Binary Output node.
Command Routing	Current behavior of the Binary Output point with select and operate requests from the master. Note: This parameter is only available in the Binary Output node.

Supported data types: `BOOL`, `USINT`

NAVIGATION BAR

Parameter	Description
Add...	<p>Opens the variable selection dialog (on page 108) to apply the zenon variables in the list.</p> <p>Note: Not active in the Frozen Counter node.</p>
Remove	<p>Removes selected variables from the list.</p> <p>A confirmation dialog is shown before a variable is deleted.</p> <p>Only active if one or more variables in the list are selected.</p> <p>Note: Not active in the Frozen Counter node.</p>
Replace variable...	<p>Opens the variable selection dialog (on page 108) to replace a variable selected in the list with another zenon variable.</p> <p>Only active if precisely one variable in the list is selected.</p>

DOUBLE BIT BINARY INPUT

You configure the parameters for selected variable(s) in the list in this area.

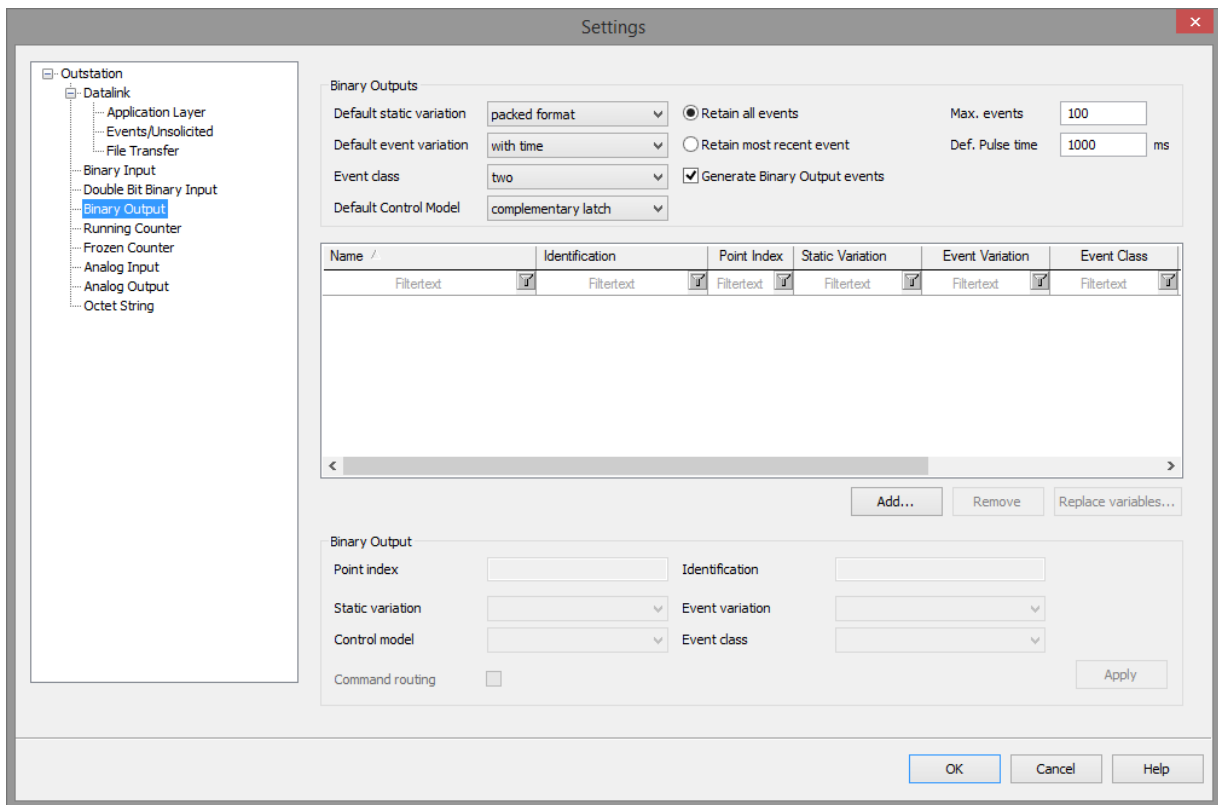
Parameter	Description
Point index	<p>This option determines the DNP3 point index in the Object Group for the selected variable. If variables are added, the point index is automatically set for the newly-added variables and numbered consecutively starting with the highest point index already present.</p> <p>A point index can only be issued once within an Object Group. The entry is validated. If a point index has already been issued, this is shown with a warning dialog.</p> <p>Note: It is recommended that you issue the point index starting with 0 without gaps. The outstation supports the maximum point index. If a higher point index is used, it should be ensured that the master also supports qualifier codes with 2 octets or 4 octets accordingly.</p>
Identification	<p>The identification is primarily taken from the variable from zenon Runtime. The identification can also be amended locally with this option.</p>
Static variation	<p>This option determines the static variation for the selected variable for the point in this Object Group. With the <code>default</code> value, the general static variation defined for this Object Group is used.</p> <p>Note: Select a static variation that is supported by the master. Not every master supports all static variations.</p> <p>Select from drop-down list:</p> <ul style="list-style-type: none"> ▶ <code>packed format</code> Object Group 3 - Variation 1 ▶ <code>with flags</code> Object Group 3 - Variation 2 <p>Default: <code>default</code></p>

Event variation	<p>This option determines the event variation for the selected variable for the point in this Object Group. With the <code>default</code> value, the general event variation defined for this Object Group is used.</p> <p>Note: Select a event variation that is supported by the master. Not every master supports all event variations.</p> <p>Select from drop-down list:</p> <ul style="list-style-type: none"> ▶ <code>without time</code> Object Group 4 - Variation 1 ▶ <code>with absolute time</code> Object Group 4 - Variation 2 ▶ <code>with relative time</code> Object Group 4 - Variation 3 <p>Default: <code>default</code></p>
Event class	<p>This option determines the event class for the selected variable for the point in this Object Group. With the <code>default</code> value, the general event class defined for this Object Group is used.</p> <p>Selection: see description for Default event class option.</p> <p>Default: <code>default</code></p>
Apply	Applies the configuration for the selected variable(s).

CLOSE DIALOG

Options	Description
OK	Applies settings and closes the dialog.
Cancel	Discards all changes and closes the dialog.
Help	Opens online help.

7.6 Binary Output



The screenshot shows the 'Settings' dialog box for the 'Binary Output' configuration. On the left is a tree view of the 'Outstation' configuration, with 'Binary Output' selected. The main area is divided into two sections: 'Binary Outputs' and 'Binary Output'.

Binary Outputs Section:

- Default static variation: packed format
- Default event variation: with time
- Event class: two
- Default Control Model: complementary latch
- Retain all events (selected)
- Retain most recent event (unselected)
- Generate Binary Output events (checked)
- Max. events: 100
- Def. Pulse time: 1000 ms

Table:

Name /	Identification	Point Index	Static Variation	Event Variation	Event Class
Filtertext	Filtertext	Filtertext	Filtertext	Filtertext	Filtertext

Binary Output Section:

- Point index: [text box]
- Identification: [text box]
- Static variation: [dropdown]
- Event variation: [dropdown]
- Control model: [dropdown]
- Event class: [dropdown]
- Command routing: ☐
- Buttons: Add..., Remove, Replace variables..., Apply

Footer: OK, Cancel, Help

You configure the DNP3 **Binary Output** points in this node.

Parameter	Description
Default static variation	<p>This option determines the default static variation for this Object Group for newly-added variables. A different static variation can be individually configured for each variable.</p> <p>Note: Select a static variation that is supported by the master. Not every master supports all static variations.</p> <p>Select from drop-down list:</p> <ul style="list-style-type: none"> ▶ packed format Object Group 10 - Variation 1 ▶ with flags Object Group 10 - Variation 2 <p>Default: packed format</p>
Event retention	<p>Behavior when forwarding value changes.</p> <p>Selection from option field.</p> <ul style="list-style-type: none"> ▶ Retain all events Each time a value, status or time stamp of a variable is changed, a new event is created and saved in the event buffer. The master gets all changes. ▶ Retain most recent event The event buffer buffer only contains one event for the last change of value, status or time stamp. The master only gets the last change and, under certain circumstances, cannot distinguish whether or how often a value has changed in the meantime. <p>Default: Retain all events</p>
Max. events	<p>Maximum number of events that are saved in the event buffer for this Object Group for all defined variables. If Retain all events is configured, this number should be at least as large as the number of configured variables.</p> <p>Default: 100</p> <p>Note: Only active if Retain all events is active.</p>

Default event variation	<p>This option determines the default event variation for this Object Group for newly-added variables. A different event variation can be individually configured for each variable.</p> <p>Note: Select a event variation that is supported by the master. Not every master supports all event variations.</p> <p>Select from drop-down list:</p> <ul style="list-style-type: none"> ▶ without time Object Group 11 - Variation 1 ▶ with time Object Group 11 - Variation 2 <p>Default: with time</p>
Event Class	<p>This option determines the default event class for this Object Group for newly-added variables. A different event class can be individually configured for each variable.</p> <p>Select from drop-down list:</p> <ul style="list-style-type: none"> ▶ none Variables are not assigned to an event class and do not generate events. No static objects are included for the variables in a response to a class 0 poll from the master. A master can only read the current value of the variables by means of a read request for the respective static object group. A master can assign the variables by means of an assign class request, not another event class. ▶ zero Variables are assigned to the event class 0, but do not generate any events. In a response to a class 0 poll from the master, the static objects are included for the variables. A master can also read the variables by means of a read request for the respective static object group. A master can assign the variables by means of an assign class request, the event class 1, 2 or 3, as a result of which the variables also generate events from this point in time. ▶ one Variables are assigned to the event class 1, and generate events. In a response to a class 0 poll from the master, the static objects are included for the variables. A master can also read the current value of the variables by means of a read request for the

	<p>respective static object group. In a response to a class 1 poll from the master , any event objects that may be present are included for the events of the variables. A master can assign the variables by means of an assign class request, the event class 0, 2 or 3. If the variable is assigned by the master of event class 0, no more events are generated.</p> <p>► two</p> <p>Variables are assigned to the event class 2, and generate events. In a response to a class 0 poll from the master, the static objects are included for the variables. A master can also read the current value of the variables by means of a read request for the respective static object group. In a response to a class 2 poll from the master , any event objects that may be present are included for the events of the variables. A master can assign the variables by means of an assign class request, the event class 0, 1 or 3. If the variable is assigned by the master of event class 0, no more events are generated.</p> <p>► three</p> <p>Variables are assigned to the event class 3, and generate events. In a response to a class 0 poll from the master, the static objects are included for the variables. A master can also read the current value of the variables by means of a read request for the respective static object group. In a response to a class 3 poll from the master , any event objects that may be present are included for the events of the variables. A master can assign the variables by means of an assign class request, the event class 0, 1 or 2. If the variable is assigned by the master of event class 0, no more events are generated.</p> <p>Default: two</p>
--	--

Default Pulse time	<p>This option determines the Default Pulse ON time.</p> <p>The variable linked for the Binary Output point is set, for the duration of the time defined here, to the value "1", "3", "4", "65" or "129" (depending on the data type of the variable for the Binary Output)</p> <p>The variable is subsequently set to the value 0 if the Master sends a Pulse ON control command with the Pulse On value "0" (default). This is applicable for Control Model with the configuration "Activation" or "complementary two output".</p> <p>If the Master sends a value for Pulse On that is different to "0" (default) with the Pulse On control command, the time from the Request is used by the Master provided this time is not more or less than five times the value of the Default Pulse time. If the value is outside, the outstation uses the Default Pulse time configured here.</p> <p>Note: The Default pulse time is not used for Control Model complementary latch.</p> <p>Default: 1000 ms</p>
Generate Binary Output events	<p>This option determines whether value changes for variables that are configured for Binary Outputs, events are also to be generated. Not all masters support events for Binary Outputs.</p> <p>Default: active</p>
Default Control Model	<p>This option determines the default Control Model for newly-added variables.</p> <p>Note: Certain combinations of Master control requests and data type (BOOL or USINT) of the variable are not possible. Also depending on whether Command Routing is active or not active for a selected variable.</p> <p>Select from drop-down list:</p> <ul style="list-style-type: none"> ► activation

	<ul style="list-style-type: none"> ▶ complementary latch ▶ complementary two output <p>Default: complementary latch</p> <p>Note: For more information, refer to the overview table in the "LATCH ON und LATCH_OFF (on page 117)" chapter in this manual.</p>
--	---

Variables taken from the zenon project are listed with their current configuration in this list.

- ▶ The list can be sorted
 - Click for the sorting on the column heading.
 - The sorting sequence is visualized with an arrow symbol next to the column heading:
 - Arrow upwards: ascending sorting
 - Arrow downwards: descending sorting
 - Another click on the column heading reverses the sorting order.
- ▶ The list can be filtered

To filter the list:

 - Enter the desired filter term in the input field below the heading. The default description of an empty field is `filter text` (shown in gray font).

Parameter	Description
Name	Variable name. The project name is placed in front of the name: ProjectName#VariableName
Identification	Identification of the variable.
Point Index	Current configuration of the DNP3 point index of the variable in this Object Group.
Static Variation	Current configuration of the static variation for the point of the variable in this Object Group.
Event Variation	Current configuration of the event variation for the point of the variable in this Object Group. Note: This parameter is not available in the Octet String node.
Event Class	Current configuration of the event class for the point of the variable in this Object Group.
Control Model	Current configuration of the Control Model . Note: This parameter is only available in the Binary Output node.
Command Routing	Current behavior of the Binary Output point with select and operate requests from the master. Note: This parameter is only available in the Binary Output node.

Supported data types: UDINT, UINT

NAVIGATION BAR

Parameter	Description
Add...	<p>Opens the variable selection dialog (on page 108) to apply the zenon variables in the list.</p> <p>Note: Not active in the Frozen Counter node.</p>
Remove	<p>Removes selected variables from the list.</p> <p>A confirmation dialog is shown before a variable is deleted.</p> <p>Only active if one or more variables in the list are selected.</p> <p>Note: Not active in the Frozen Counter node.</p>
Replace variable...	<p>Opens the variable selection dialog (on page 108) to replace a variable selected in the list with another zenon variable.</p> <p>Only active if precisely one variable in the list is selected.</p>

BINARY OUTPUT

You configure the parameters for selected variable(s) in the list in this area.

Parameter	Description
Point index	<p>This option determines the DNP3 point index in the Object Group for the selected variable. If variables are added, the point index is automatically set for the newly-added variables and numbered consecutively starting with the highest point index already present.</p> <p>A point index can only be issued once within an Object Group. The entry is validated. If a point index has already been issued, this is shown with a warning dialog.</p> <p>Note: It is recommended that you issue the point index starting with 0 without gaps. The outstation supports the maximum point index. If a higher point index is used, it should be ensured that the master also supports qualifier codes with 2 octets or 4 octets accordingly.</p>
Identification	<p>The identification is primarily taken from the variable from zenon Runtime. The identification can also be amended locally with this option.</p>
Static variation	<p>This option determines the static variation for the selected variable for the point in this Object Group. With the <code>default</code> value, the general static variation defined for this Object Group is used.</p> <p>Note: Select a static variation that is supported by the master. Not every master supports all static variations.</p> <p>Select from drop-down list:</p> <ul style="list-style-type: none"> ▸ <code>packed format</code> Object Group 10 - Variation 1 ▸ <code>with flags</code> Object Group 10 - Variation 2 <p>Default: <code>default</code></p>

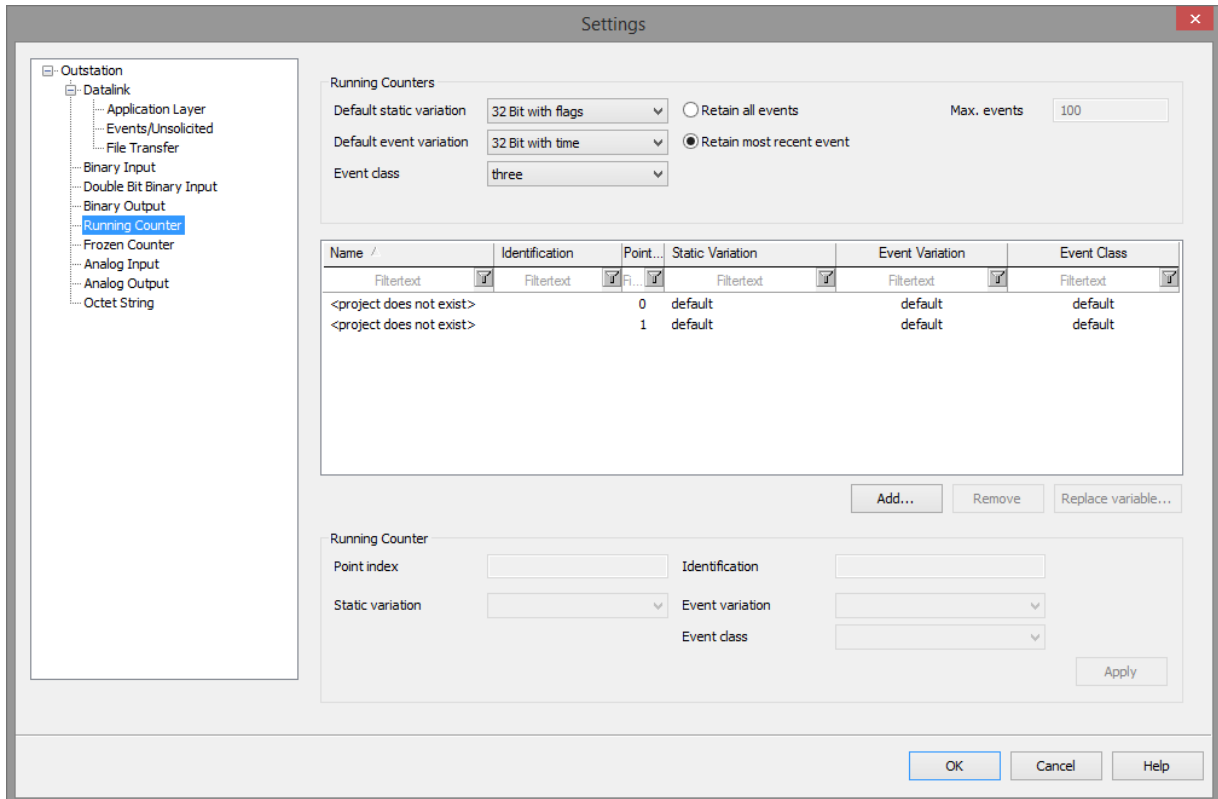
Event variation	<p>This option determines the event variation for the selected variable for the point in this Object Group. With the <code>default</code> value, the general event variation defined for this Object Group is used.</p> <p>Note: Select a event variation that is supported by the master. Not every master supports all event variations.</p> <p>Select from drop-down list:</p> <ul style="list-style-type: none"> ▶ <code>without time</code> Object Group 11 - Variation 1 ▶ <code>with time</code> Object Group 11 - Variation 2 <p>Default: <code>default</code></p>
Control model	<p>This option determines the Control Model for the selected variable.</p> <p>Note: Certain combinations of Master control requests and data type (BOOL or USINT) of the variable are not possible. Also depending on whether Command Routing is active or not active for a selected variable.</p> <p>Select from drop-down list:</p> <ul style="list-style-type: none"> ▶ <code>activation</code> Object Group n - Variation n ▶ <code>complementary latch</code> Object Group n - Variation n ▶ <code>complementary two output</code> Object Group n - Variation n <p>Default: <code>default</code></p> <p>You can find detailed information on this in the "Command processing - Command Routing" (on page 114) chapter.</p>

Event class	<p>This option determines the event class for the selected variable for the point in this Object Group. With the <code>default</code> value, the general event class defined for this Object Group is used.</p> <p>Selection: see description for Default event class option.</p> <p>Default: <code>default</code></p>
Command routing	<p>This option determines the behavior of Binary Output point for the selected variable for select and operate requests from the master.</p> <ul style="list-style-type: none"> ▶ Active: select requests from the master and operate requests from the master are routed further to the command input in zenon Runtime. Only once a response to the select or operate has been received by the PLC (to the command input in Runtime), a corresponding positive, negative select response OR operate response is sent to the master by the outstation. Direct operate requests Are immediately responded to positively by the Outstation if the linked variable has been written in Runtime. ▶ Inactive: select requests Is always responded to immediately by the Outstation if a variable has been configured for the Binary Output point and the variable is present in Runtime. operate requests and direct operate requests are immediately responded to positively by the outstation if the linked variable has been written in Runtime. <p>Default: <code>active</code></p> <p>You can find further details on Command routing in the "Select before Operate (on page 114)" chapter.</p>
Apply	Applies the configuration for the selected variable(s).

CLOSE DIALOG

Options	Description
OK	Applies settings and closes the dialog.
Cancel	Discards all changes and closes the dialog.
Help	Opens online help.

7.7 Running Counter



The screenshot shows the 'Settings' dialog box for the 'Running Counter' node. The left sidebar shows a tree view with 'Outstation' expanded, and 'Running Counter' selected under 'DataLink'. The main area is divided into two sections: 'Running Counters' and 'Running Counter'.

Running Counters Section:

- Default static variation: 32 Bit with flags
- Default event variation: 32 Bit with time
- Event class: three
- Retain all events: ☐
- Retain most recent event: ☒
- Max. events: 100

Running Counter Table:

Name /	Identification	Point...	Static Variation	Event Variation	Event Class
<project does not exist>		0	default	default	default
<project does not exist>		1	default	default	default

Running Counter Section:

- Point index:
- Static variation:
- Identification:
- Event variation:
- Event class:

Buttons: Add..., Remove, Replace variable..., Apply, OK, Cancel, Help.

You configure the DNP3 **Running counter points** in this node.

RUNNING COUNTER

Parameter	Description
Default static variation	<p>This option determines the event variation for the selected variable for the point in this Object Group. With the <code>default</code> value, the general event variation defined for this Object Group is used.</p> <p>Note: Select a event variation that is supported by the master. Not every master supports all event variations.</p> <p>Select from drop-down list:</p> <ul style="list-style-type: none"> ▶ 32 Bit with flags Object Group 20 - Variation 1 ▶ 16 Bit with flags Object Group 20 - Variation 2 ▶ 32 Bit without flags Object Group 20 - Variation 5 ▶ 16 Bit without flags Object Group 20 - Variation 6 <p>Default: 32 Bit with flags</p>
Event retention	<p>Behavior when forwarding value changes.</p> <p>Selection from option field.</p> <ul style="list-style-type: none"> ▶ Retain all events Each time a value, status or time stamp of a variable is changed, a new event is created and saved in the event buffer. The master gets all changes. ▶ Retain most recent event The event buffer buffer only contains one event for the last change of value, status or time stamp. The master only gets the last change and, under certain circumstances, cannot distinguish whether or how often a value has changed in the meantime. <p>Default: Retain most recent event</p>
Max. events	<p>Maximum number of events that are saved in the event buffer for this Object Group for all defined variables. If <code>Retain all events</code> is configured, this number should be at least as large as the number of configured variables.</p> <p>Default: 100</p>

	<p>Note: Only active if Retain all events is active.</p>
Default event variation	<p>This option determines the default event variation for this Object Group for newly-added variables. A different event variation can be individually configured for each variable.</p> <p>Note: Select a event variation that is supported by the master. Not every master supports all event variations.</p> <p>Select from drop-down list:</p> <ul style="list-style-type: none"> ▶ 32 Bit with flags Object Group 22 - Variation 1 ▶ 16 Bit with flags Object Group 22 - Variation 2 ▶ 32 Bit with time Object Group 22 - Variation 5 ▶ 16 Bit with time Object Group 22 - Variation 6 <p>Default: 32 Bit with time</p>
Event class	<p>This option determines the default event class for this Object Group for newly-added variables. A different event class can be individually configured for each variable.</p> <p>Select from drop-down list:</p> <ul style="list-style-type: none"> ▶ none Variables are not assigned to an event class and do not generate events. No static objects are included for the variables in a response to a class 0 poll from the master. A master can only read the current value of the variables by means of a read request for the respective static object group. A master can assign the variables by means of an assign class request, not another event class. ▶ zero Variables are assigned to the event class 0, but do not generate any events. In a response to a class 0 poll from the master, the static objects are included for the variables. A master can also read the variables by means of a read request for the respective static object group. A master can assign the variables by means of an assign class request, the event class 1, 2 or 3, as a result of which the variables also generate events from this point in time.

	<p>► one</p> <p>Variables are assigned to the event class 1, and generate events. In a response to a class 0 poll from the master, the static objects are included for the variables. A master can also read the current value of the variables by means of a read request for the respective static object group. In a response to a class 1 poll from the master, any event objects that may be present are included for the events of the variables. A master can assign the variables by means of an assign class request, the event class 0, 2 or 3. If the variable is assigned by the master of event class 0, no more events are generated.</p> <p>► two</p> <p>Variables are assigned to the event class 2, and generate events. In a response to a class 0 poll from the master, the static objects are included for the variables. A master can also read the current value of the variables by means of a read request for the respective static object group. In a response to a class 2 poll from the master, any event objects that may be present are included for the events of the variables. A master can assign the variables by means of an assign class request, the event class 0, 1 or 3. If the variable is assigned by the master of event class 0, no more events are generated.</p> <p>► three</p> <p>Variables are assigned to the event class 3, and generate events. In a response to a class 0 poll from the master, the static objects are included for the variables. A master can also read the current value of the variables by means of a read request for the respective static object group. In a response to a class 3 poll from the master, any event objects that may be present are included for the events of the variables. A master can assign the variables by means of an assign class request, the event class 0, 1 or 2. If the variable is assigned by the master of event class 0, no more events are generated.</p> <p>Default: three</p>
--	---

Variables taken from the zenon project are listed with their current configuration in this list.

- The list can be sorted
 - Click for the sorting on the column heading.
 - The sorting sequence is visualized with an arrow symbol next to the column heading:
 - Arrow upwards: ascending sorting
 - Arrow downwards: descending sorting

- Another click on the column heading reverses the sorting order.
- The list can be filtered
To filter the list:
 - Enter the desired filter term in the input field below the heading. The default description of an empty field is `filter text` (shown in gray font).

Parameter	Description
Name	Variable name. The project name is placed in front of the name: <code>ProjectName#VariableName</code>
Identification	Identification of the variable.
Point Index	Current configuration of the DNP3 point index of the variable in this Object Group.
Static Variation	Current configuration of the static variation for the point of the variable in this Object Group.
Event Variation	Current configuration of the event variation for the point of the variable in this Object Group. Note: This parameter is not available in the Octet String node.
Event Class	Current configuration of the event class for the point of the variable in this Object Group.
Control Model	Current configuration of the Control Model . Note: This parameter is only available in the Binary Output node.
Command Routing	Current behavior of the Binary Output point with select and operate requests from the master. Note: This parameter is only available in the Binary Output node.

Supported data types: UDINT, UINT

NAVIGATION BAR

Parameter	Description
Add...	<p>Opens the variable selection dialog (on page 108) to apply the zenon variables in the list.</p> <p>Note: Not active in the Frozen Counter node.</p>
Remove	<p>Removes selected variables from the list.</p> <p>A confirmation dialog is shown before a variable is deleted.</p> <p>Only active if one or more variables in the list are selected.</p> <p>Note: Not active in the Frozen Counter node.</p>
Replace variable...	<p>Opens the variable selection dialog (on page 108) to replace a variable selected in the list with another zenon variable.</p> <p>Only active if precisely one variable in the list is selected.</p>

RUNNING COUNTER

You configure the parameters for selected variable(s) in the list in this area.

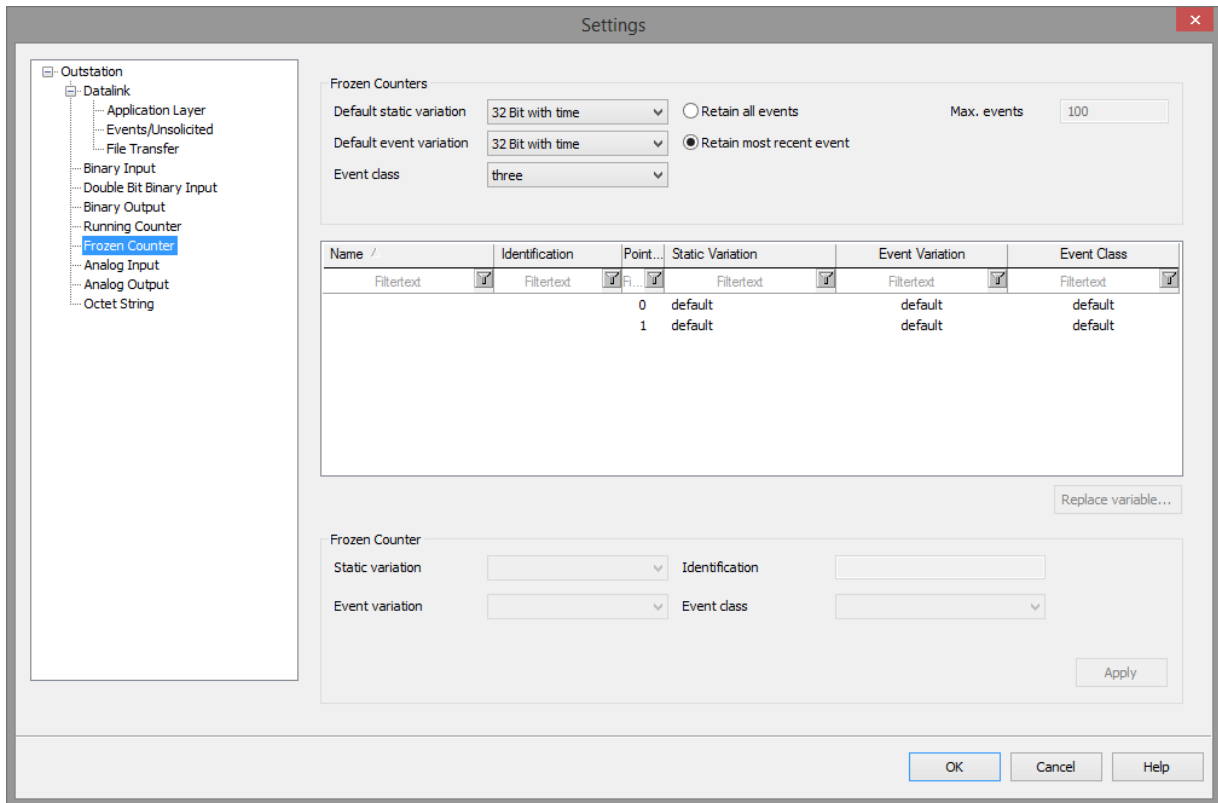
Parameter	Description
Point index	<p>This option determines the DNP3 point index in the Object Group for the selected variable. If variables are added, the point index is automatically set for the newly-added variables and numbered consecutively starting with the highest point index already present.</p> <p>A point index can only be issued once within an Object Group. The entry is validated. If a point index has already been issued, this is shown with a warning dialog.</p> <p>Note: It is recommended that you issue the point index starting with 0 without gaps. The outstation supports the maximum point index. If a higher point index is used, it should be ensured that the master also supports qualifier codes with 2 octets or 4 octets accordingly.</p>
Identification	<p>The identification is primarily taken from the variable from zenon Runtime. The identification can also be amended locally with this option.</p>
Static variation	<p>This option determines the static variation for the selected variable for the point in this Object Group. With the <code>default</code> value, the general static variation defined for this Object Group is used.</p> <p>Note: Select a static variation that is supported by the master. Not every master supports all static variations.</p> <p>Select from drop-down list:</p> <ul style="list-style-type: none"> ▶ 32 Bit with flags Object Group 20 - Variation 1 ▶ 16 Bit with flags Object Group 20 - Variation 2 ▶ 32 Bit without flags Object Group 20 - Variation 5 ▶ 16 Bit without flags Object Group 20 - Variation 6 <p>Default: <code>default</code></p>
Event variation	<p>This option determines the event variation for the selected variable for the point in this Object Group. With the <code>default</code> value, the general event variation defined for this Object Group is used.</p>

	<p>Note: Select a event variation that is supported by the master. Not every master supports all event variations.</p> <p>Select from drop-down list:</p> <ul style="list-style-type: none"> ▶ 32 Bit with flags Object Group 22 - Variation 1 ▶ 16 Bit with flags Object Group 22 - Variation 2 ▶ 32 Bit with time Object Group 22 - Variation 5 ▶ 16 Bit with time Object Group 22 - Variation 6 <p>Default: default</p>
Event class	<p>This option determines the event class for the selected variable for the point in this Object Group. With the <code>default</code> value, the general event class defined for this Object Group is used.</p> <p>Selection: see description for Default event class option.</p> <p>Default: default</p>
Apply	Applies the configuration for the selected variable(s).

CLOSE DIALOG

Options	Description
OK	Applies settings and closes the dialog.
Cancel	Discards all changes and closes the dialog.
Help	Opens online help.

7.8 Frozen Counter



The screenshot shows the 'Settings' dialog for the 'Frozen Counter' node. On the left is a tree view of the project structure, with 'Frozen Counter' selected under 'Outstation'. The main area is divided into two sections: 'Frozen Counters' and 'Frozen Counter'.

Frozen Counters Section:

- Default static variation: 32 Bit with time
- Default event variation: 32 Bit with time
- Event class: three
- Retain all events: ☐
- Retain most recent event: ☒
- Max. events: 100

Frozen Counter Table:

Name /	Identification	Point...	Static Variation	Event Variation	Event Class
Filtertext	Filtertext	Filtertext	Filtertext	Filtertext	Filtertext
		0	default	default	default
		1	default	default	default

Frozen Counter Section:

- Static variation: [dropdown]
- Identification: [text box]
- Event variation: [dropdown]
- Event class: [dropdown]

Buttons: OK, Cancel, Help, Apply, Replace variable...

You configure the DNP3 Frozen counter points in this node.

Parameter	Description
Default static variation	<p>This option determines the default static variation for this Object Group for newly-added variables. A different static variation can be individually configured for each variable.</p> <p>Note: Select a static variation that is supported by the master. Not every master supports all static variations.</p> <p>Select from drop-down list:</p> <ul style="list-style-type: none"> ▶ 32 Bit with flags Object Group 21 - Variation 1 ▶ 16 Bit with flags Object Group 21 - Variation 2 ▶ 32 Bit with time Object Group 21 - Variation 5 ▶ 16 Bit with time Object Group 21 - Variation 6 ▶ 32 Bit without flags Object Group 21 - Variation 9 ▶ 16 Bit without flags Object Group 21 - Variation 10 <p>Default: 32 Bit with time</p>
Event retention	<p>Behavior when forwarding value changes.</p> <p>Selection from option field.</p> <ul style="list-style-type: none"> ▶ Retain all events Each time a value, status or time stamp of a variable is changed, a new event is created and saved in the event buffer. The master gets all changes. ▶ Retain most recent event The event buffer buffer only contains one event for the last change of value, status or time stamp. The master only gets the last change and, under certain circumstances, cannot distinguish whether or how often a value has changed in the meantime. <p>Default: Retain most recent event</p>
Max events	<p>Maximum number of events that are saved in the event buffer for this Object Group for all defined variables. If Retain all events is configured, this number should</p>

	<p>be at least as large as the number of configured variables.</p> <p>Default: 100</p> <p>Note: Only active if Retain all events is active.</p>
Default event variations	<p>This option determines the default event variation for this Object Group for newly-added variables. A different event variation can be individually configured for each variable.</p> <p>Note: Select a event variation that is supported by the master. Not every master supports all event variations.</p> <p>Select from drop-down list:</p> <ul style="list-style-type: none"> ▶ 32 Bit with flags Object Group 23 - Variation 1 ▶ 16 Bit with flags Object Group 23 - Variation 2 ▶ 32 Bit with time Object Group 23 - Variation 5 ▶ 16 Bit with time Object Group 23 - Variation 6 <p>Default: 32 Bit with time</p>
Event class	<p>This option determines the default event class for this Object Group for newly-added variables. A different event class can be individually configured for each variable.</p> <p>Select from drop-down list:</p> <ul style="list-style-type: none"> ▶ none Variables are not assigned to an event class and do not generate events. No static objects are included for the variables in a response to a class 0 poll from the master. A master can only read the current value of the variables by means of a read request for the respective static object group. A master can assign the variables by means of an assign class request, not another event class. ▶ zero Variables are assigned to the event class 0, but do not generate any events. In a response to a class 0 poll from the master, the static objects are included for the variables. A master can also read the variables by means of a read request for the respective static object group. A master can assign the variables by means of an assign class request,

	<p>the event class 1, 2 or 3, as a result of which the variables also generate events from this point in time.</p> <ul style="list-style-type: none"> ▶ <code>one</code> Variables are assigned to the event class 1, and generate events. In a response to a class 0 poll from the master, the static objects are included for the variables. A master can also read the current value of the variables by means of a read request for the respective static object group. In a response to a class 1 poll from the master, any event objects that may be present are included for the events of the variables. A master can assign the variables by means of an assign class request, the event class 0, 2 or 3. If the variable is assigned by the master of event class 0, no more events are generated. ▶ <code>two</code> Variables are assigned to the event class 2, and generate events. In a response to a class 0 poll from the master, the static objects are included for the variables. A master can also read the current value of the variables by means of a read request for the respective static object group. In a response to a class 2 poll from the master, any event objects that may be present are included for the events of the variables. A master can assign the variables by means of an assign class request, the event class 0, 1 or 3. If the variable is assigned by the master of event class 0, no more events are generated. ▶ <code>three</code> Variables are assigned to the event class 3, and generate events. In a response to a class 0 poll from the master, the static objects are included for the variables. A master can also read the current value of the variables by means of a read request for the respective static object group. In a response to a class 3 poll from the master, any event objects that may be present are included for the events of the variables. A master can assign the variables by means of an assign class request, the event class 0, 1 or 2. If the variable is assigned by the master of event class 0, no more events are generated. <p>Default: <code>three</code></p>
--	--

Variables taken from the zenon project are listed with their current configuration in this list.

- ▶ The list can be sorted
 - Click for the sorting on the column heading.

- The sorting sequence is visualized with an arrow symbol next to the column heading:
 Arrow upwards: ascending sorting
 Arrow downwards: descending sorting
- Another click on the column heading reverses the sorting order.
- ▶ The list can be filtered
 To filter the list:
 - Enter the desired filter term in the input field below the heading. The default description of an empty field is `filter text` (shown in gray font).

Parameter	Description
Name	Variable name. The project name is placed in front of the name: <code>ProjectName#VariableName</code>
Identification	Identification of the variable.
Point Index	Current configuration of the DNP3 point index of the variable in this Object Group.
Static Variation	Current configuration of the static variation for the point of the variable in this Object Group.
Event Variation	Current configuration of the event variation for the point of the variable in this Object Group. Note: This parameter is not available in the Octet String node.
Event Class	Current configuration of the event class for the point of the variable in this Object Group.
Control Model	Current configuration of the Control Model . Note: This parameter is only available in the Binary Output node.
Command Routing	Current behavior of the Binary Output point with select and operate requests from the master. Note: This parameter is only available in the Binary Output node.

Supported data types: UDINT, UINT

NAVIGATION BAR

Parameter	Description
Add...	<p>Opens the variable selection dialog (on page 108) to apply the zenon variables in the list.</p> <p>Note: Not active in the Frozen Counter node.</p>
Remove	<p>Removes selected variables from the list.</p> <p>A confirmation dialog is shown before a variable is deleted.</p> <p>Only active if one or more variables in the list are selected.</p> <p>Note: Not active in the Frozen Counter node.</p>
Replace variable...	<p>Opens the variable selection dialog (on page 108) to replace a variable selected in the list with another zenon variable.</p> <p>Only active if precisely one variable in the list is selected.</p>

FROZEN COUNTER

You configure the parameters for selected variable(s) in the list in this area.

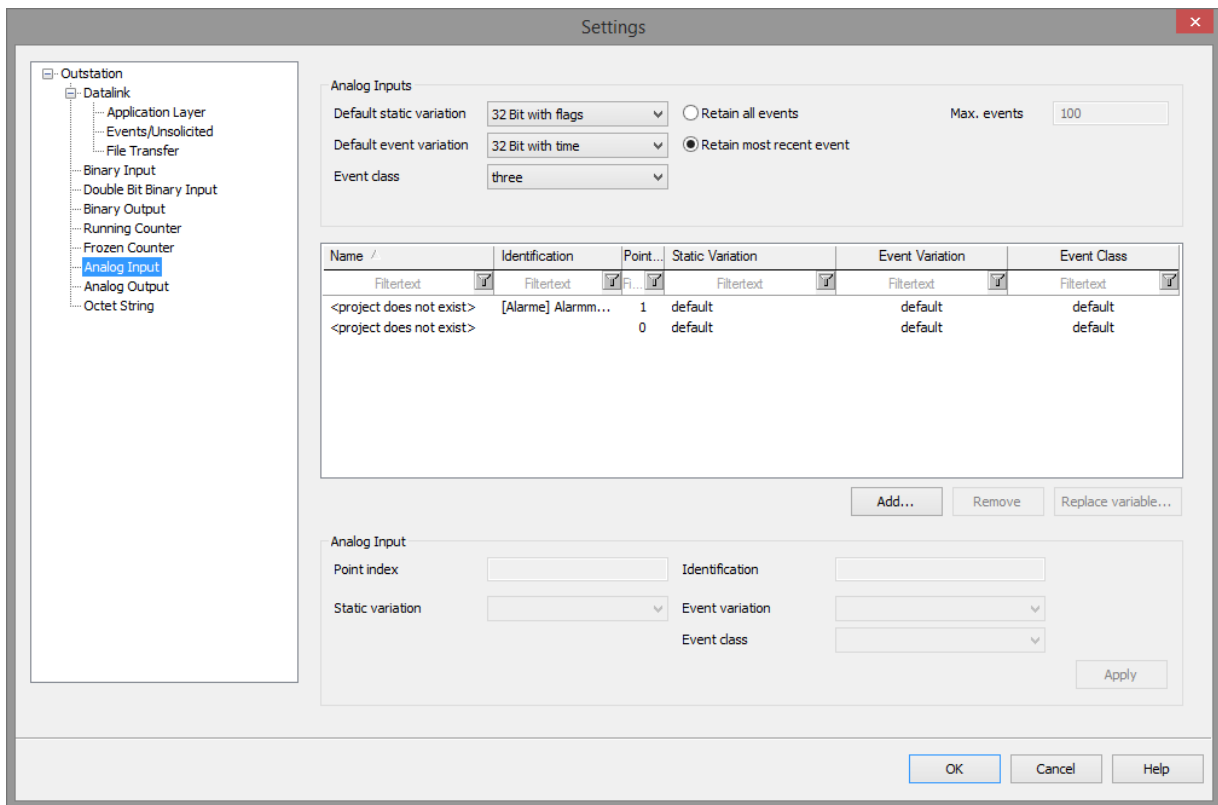
Parameter	Description
Static variation	<p>This option determines the static variation for the selected variable for the point in this Object Group. With the <code>default</code> value, the general static variation defined for this Object Group is used.</p> <p>Note: Select a static variation that is supported by the master. Not every master supports all static variations.</p> <p>Select from drop-down list:</p> <ul style="list-style-type: none"> ▶ 32 Bit with flags Object Group 21 - Variation 1 ▶ 16 Bit with flags Object Group 21 - Variation 2 ▶ 32 Bit with time Object Group 21 - Variation 5 ▶ 16 Bit with time Object Group 21 - Variation 6 ▶ 32 Bit without flags Object Group 21 - Variation 9 ▶ 16 Bit without flags Object Group 21 - Variation 10 <p>Default: <code>default</code></p>
Identification	<p>The identification is primarily taken from the variable from zenon Runtime. The identification can also be amended locally with this option.</p>
Event variation	<p>This option determines the event variation for the selected variable for the point in this Object Group. With the <code>default</code> value, the general event variation defined for this Object Group is used.</p> <p>Note: Select a event variation that is supported by the master. Not every master supports all event variations.</p> <p>Select from drop-down list:</p> <ul style="list-style-type: none"> ▶ 32 Bit with flags Object Group 23 - Variation 1 ▶ 16 Bit with flags Object Group 23 - Variation 2 ▶ w32 Bit with time Object Group 23 - Variation 5

	▶ 16 Bit with time Object Group 23 - Variation 6 Default: default
Event class	<p>This option determines the event class for the selected variable for the point in this Object Group. With the <code>default</code> value, the general event class defined for this Object Group is used.</p> <p>Selection: see description for Default event class option.</p> <p>Default: default</p>
Apply	Applies the configuration for the selected variable(s).

CLOSE DIALOG

Options	Description
OK	Applies settings and closes the dialog.
Cancel	Discards all changes and closes the dialog.
Help	Opens online help.

7.9 Analog Input



You configure the DNP3 Analog Input points in this node.

Parameter	Description
Default static variation	<p>This option determines the default static variation for this Object Group for newly-added variables. A different static variation can be individually configured for each variable.</p> <p>Note: Select a static variation that is supported by the master. Not every master supports all static variations.</p> <p>Select from drop-down list:</p> <ul style="list-style-type: none"> ▶ 32 Bit with flags Object Group 30 - Variation 1 ▶ 16 Bit with flags Object Group 30 - Variation 2 ▶ 32 Bit without flags Object Group 30 - Variation 3 ▶ 16 Bit without flags Object Group 30 - Variation 4 ▶ Real with flags (Single-precision with flag) Object Group 30 - Variation 5 ▶ LReal with flags (Double-precision with flag) Object Group 30 - Variation 6 <p>Default: 32 Bit with flags</p>
Event retention	<p>Behavior when forwarding value changes.</p> <p>Selection from option field.</p> <ul style="list-style-type: none"> ▶ Retain all events Each time a value, status or time stamp of a variable is changed, a new event is created and saved in the event buffer. The master gets all changes. ▶ Retain most recent event The event buffer buffer only contains one event for the last change of value, status or time stamp. The master only gets the last change and, under certain circumstances, cannot distinguish whether or how often a value has changed in the meantime. <p>Default: Retain most recent event</p>
Max. events	Maximum number of events that are saved in the event

	<p>buffer for this Object Group for all defined variables. If <code>Retain all events</code> is configured, this number should be at least as large as the number of configured variables.</p> <p>Default: 100</p> <p>Note: Only active if <code>Retain all events</code> is active.</p>
Default event variation	<p>This option determines the default event variation for this Object Group for newly-added variables. A different event variation can be individually configured for each variable.</p> <p>Note: Select a event variation that is supported by the master. Not every master supports all event variations.</p> <p>Select from drop-down list:</p> <ul style="list-style-type: none"> ▶ 32 Bit without time Object Group 32 - Variation 1 ▶ 16 Bit without time Object Group 32 - Variation 2 ▶ 32 Bit with time Object Group 32 - Variation 3 ▶ 16 Bit with time Object Group 32 - Variation 4 ▶ Real without time (Single-precision without time) Object Group 32 - Variation 5 ▶ LReal without time (Double-precision without time) Object Group 32 - Variation 6 ▶ Real with time (Single-precision with time) Object Group 32 - Variation 7 ▶ LReal with time (Double-precision with time) Object Group 32 - Variation 8 <p>Default: 32 Bit with time</p>
Event class	<p>This option determines the default event class for this Object Group for newly-added variables. A different event class can be individually configured for each variable.</p> <p>Select from drop-down list:</p> <ul style="list-style-type: none"> ▶ none Variables are not assigned to an event class and do not generate events. No static objects are included

	<p>for the variables in a response to a class 0 poll from the master. A master can only read the current value of the variables by means of a read request for the respective static object group. A master can assign the variables by means of an assign class request, not another event class.</p> <ul style="list-style-type: none"> ▶ zero Variables are assigned to the event class 0, but do not generate any events. In a response to a class 0 poll from the master, the static objects are included for the variables. A master can also read the variables by means of a read request for the respective static object group. A master can assign the variables by means of an assign class request, the event class 1, 2 or 3, as a result of which the variables also generate events from this point in time. ▶ one Variables are assigned to the event class 1, and generate events. In a response to a class 0 poll from the master, the static objects are included for the variables. A master can also read the current value of the variables by means of a read request for the respective static object group. In a response to a class 1 poll from the master, any event objects that may be present are included for the events of the variables. A master can assign the variables by means of an assign class request, the event class 0, 2 or 3. If the variable is assigned by the master of event class 0, no more events are generated. ▶ two Variables are assigned to the event class 2, and generate events. In a response to a class 0 poll from the master, the static objects are included for the variables. A master can also read the current value of the variables by means of a read request for the respective static object group. In a response to a class 2 poll from the master, any event objects that may be present are included for the events of the variables. A master can assign the variables by means of an assign class request, the event class 0, 1 or 3. If the variable is assigned by the master of event class 0, no more events are generated. ▶ three Variables are assigned to the event class 3, and generate events. In a response to a class 0 poll from the master, the static objects are included for the variables. A master can also read the current value of the variables by means of a read request for the
--	---

	<p>respective static object group. In a response to a class 3 poll from the master , any event objects that may be present are included for the events of the variables. A master can assign the variables by means of an assign class request, the event class 0, 1 or 2. If the variable is assigned by the master of event class 0, no more events are generated.</p> <p>Default: <code>three</code></p>
--	---

Variables taken from the zenon project are listed with their current configuration in this list.

► The list can be sorted

- Click for the sorting on the column heading.
- The sorting sequence is visualized with an arrow symbol next to the column heading:
 Arrow upwards: ascending sorting
 Arrow downwards: descending sorting
- Another click on the column heading reverses the sorting order.

► The list can be filtered

To filter the list:

- Enter the desired filter term in the input field below the heading. The default description of an empty field is `filter text` (shown in gray font).

Parameter	Description
Name	Variable name. The project name is placed in front of the name: ProjectName#VariableName
Identification	Identification of the variable.
Point Index	Current configuration of the DNP3 point index of the variable in this Object Group.
Static Variation	Current configuration of the static variation for the point of the variable in this Object Group.
Event Variation	Current configuration of the event variation for the point of the variable in this Object Group. Note: This parameter is not available in the Octet String node.
Event Class	Current configuration of the event class for the point of the variable in this Object Group.
Control Model	Current configuration of the Control Model . Note: This parameter is only available in the Binary Output node.
Command Routing	Current behavior of the Binary Output point with select and operate requests from the master. Note: This parameter is only available in the Binary Output node.

Supported data types: BOOL, Byte, DINT, DWORD, Date, Date_and_Time, INT, LINT, LREAL, LWORD, REAL, SINT, TIME, TOD, UDINT, UINT, ULINT, USINT, WORD

NAVIGATION BAR

Parameter	Description
Add...	<p>Opens the variable selection dialog (on page 108) to apply the zenon variables in the list.</p> <p>Note: Not active in the Frozen Counter node.</p>
Remove	<p>Removes selected variables from the list.</p> <p>A confirmation dialog is shown before a variable is deleted.</p> <p>Only active if one or more variables in the list are selected.</p> <p>Note: Not active in the Frozen Counter node.</p>
Replace variable...	<p>Opens the variable selection dialog (on page 108) to replace a variable selected in the list with another zenon variable.</p> <p>Only active if precisely one variable in the list is selected.</p>

ANALOG INPUT

You configure the parameters for selected variable(s) in the list in this area.

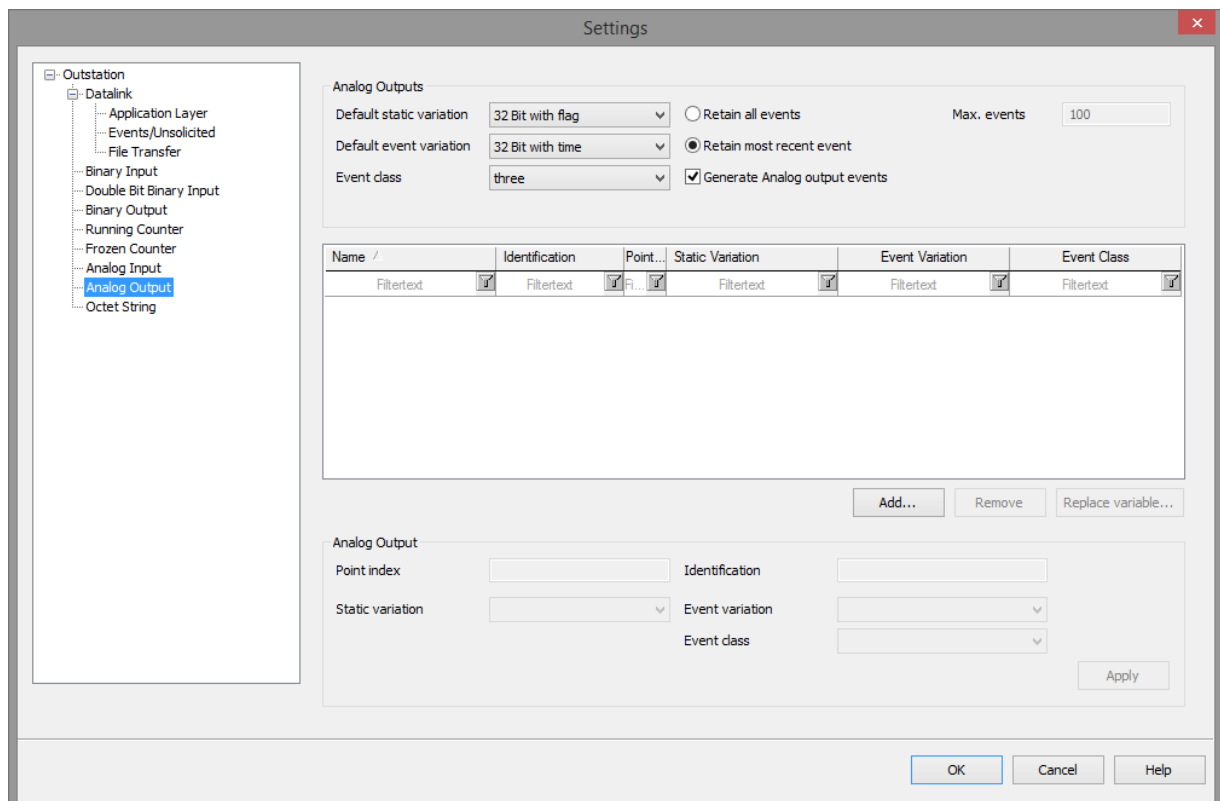
Parameter	Description
Point index	<p>This option determines the DNP3 point index in the Object Group for the selected variable. If variables are added, the point index is automatically set for the newly-added variables and numbered consecutively starting with the highest point index already present.</p> <p>A point index can only be issued once within an Object Group. The entry is validated. If a point index has already been issued, this is shown with a warning dialog.</p> <p>Note: It is recommended that you issue the point index starting with 0 without gaps. The outstation supports the maximum point index. If a higher point index is used, it should be ensured that the master also supports qualifier codes with 2 octets or 4 octets accordingly.</p>
Identification	<p>The identification is primarily taken from the variable from zenon Runtime. The identification can also be amended locally with this option.</p>
Static variation	<p>This option determines the static variation for the selected variable for the point in this Object Group. With the <code>default</code> value, the general static variation defined for this Object Group is used.</p> <p>Note: Select a static variation that is supported by the master. Not every master supports all static variations.</p> <p>Select from drop-down list:</p> <ul style="list-style-type: none"> ▶ 32 Bit with flags Object Group 30 - Variation 1 ▶ 16 Bit with flags Object Group 30 - Variation 2 ▶ 32 Bit without flags Object Group 30 - Variation 3 ▶ 16 Bit without flags Object Group 30 - Variation 4 ▶ Real with flags (Single-precision with flag) Object Group 30 - Variation 5 ▶ LReal with flags (Double-precision with flag) Object Group 30 - Variation 6

	Default: default
Event variation	<p>This option determines the event variation for the selected variable for the point in this Object Group. With the <code>default</code> value, the general event variation defined for this Object Group is used.</p> <p>Note: Select a event variation that is supported by the master. Not every master supports all event variations.</p> <p>Select from drop-down list:</p> <ul style="list-style-type: none"> ▶ 32 Bit without time Object Group 32 - Variation 1 ▶ 16 Bit without time Object Group 32 - Variation 2 ▶ 32 Bit with time Object Group 32 - Variation 3 ▶ 16 Bit with time Object Group 32 - Variation 4 ▶ Real without time (Single-precision without time) Object Group 32 - Variation 5 ▶ LReal without time (Double-precision without time) Object Group 32 - Variation 6 ▶ Real with time (Single-precision with time) Object Group 32 - Variation 7 ▶ LReal with time (Double-precision with time) Object Group 32 - Variation 8 <p>Default: default</p>
Event class	<p>This option determines the event class for the selected variable for the point in this Object Group. With the <code>default</code> value, the general event class defined for this Object Group is used.</p> <p>Selection: see description for Default event class option.</p> <p>Default: default</p>
Apply	Applies the configuration for the selected variable(s).

CLOSE DIALOG

Options	Description
OK	Applies settings and closes the dialog.
Cancel	Discards all changes and closes the dialog.
Help	Opens online help.

7.10 Analog Output



You configure the **DNP3 Analog Output points** in this node.

Parameter	Description
Default static variation	<p>This option determines the default static variation for this Object Group for newly-added variables. A different static variation can be individually configured for each variable.</p> <p>Note: Select a static variation that is supported by the master. Not every master supports all static variations.</p> <p>Select from drop-down list:</p> <ul style="list-style-type: none"> ▶ 32 Bit with flag Object Group 40 - Variation 1 ▶ 16 Bit with flag Object Group 40 - Variation 2 ▶ Real with flag (Single-precision with flag) Object Group 40 - Variation 3 ▶ LReal with flag (Double-precision with flag) Object Group 40 - Variation 4 <p>Default: 32 Bit with flag</p>
Event retention	<p>Behavior when forwarding value changes.</p> <p>Selection from option field.</p> <ul style="list-style-type: none"> ▶ Retain all events Each time a value, status or time stamp of a variable is changed, a new event is created and saved in the event buffer. The master gets all changes. ▶ Retain most recent event The event buffer buffer only contains one event for the last change of value, status or time stamp. The master only gets the last change and, under certain circumstances, cannot distinguish whether or how often a value has changed in the meantime. <p>Default: Retain most recent event</p>
Max events	<p>Maximum number of events that are saved in the event buffer for this Object Group for all defined variables. If Retain all events is configured, this number should be at least as large as the number of configured variables.</p> <p>Default: 100</p>

	<p>Note: Only active if Retain all events is active.</p>
Default event variations	<p>This option determines the default event variation for this Object Group for newly-added variables. A different event variation can be individually configured for each variable.</p> <p>Note: Select a event variation that is supported by the master. Not every master supports all event variations.</p> <p>Select from drop-down list:</p> <ul style="list-style-type: none"> ▶ 32 Bit without time Object Group 42 - Variation 1 ▶ 16 Bit without time Object Group 42 - Variation 2 ▶ 32 Bit with time Object Group 42 - Variation 3 ▶ 16 Bit with time Object Group 42 - Variation 4 ▶ Real without time (Single-precision without time) Object Group 42 - Variation 5 ▶ LReal without time (Double-precision without time) Object Group 42 - Variation 6 ▶ Real with time (Double-precision with time) Object Group 42 - Variation 7 ▶ LReal with time (Double-precision with time) Object Group 42 - Variation 8 <p>Default: 32 Bit with time</p>
Event class	<p>This option determines the default event class for this Object Group for newly-added variables. A different event class can be individually configured for each variable.</p> <p>Select from drop-down list:</p> <ul style="list-style-type: none"> ▶ none Variables are not assigned to an event class and do not generate events. No static objects are included for the variables in a response to a class 0 poll from the master. A master can only read the current value of the variables by means of a read request for the respective static object group. A master can assign

	<p>the variables by means of an assign class request, not another event class.</p> <ul style="list-style-type: none"> ▶ zero Variables are assigned to the event class 0, but do not generate any events. In a response to a class 0 poll from the master, the static objects are included for the variables. A master can also read the variables by means of a read request for the respective static object group. A master can assign the variables by means of an assign class request, the event class 1, 2 or 3, as a result of which the variables also generate events from this point in time. ▶ one Variables are assigned to the event class 1, and generate events. In a response to a class 0 poll from the master, the static objects are included for the variables. A master can also read the current value of the variables by means of a read request for the respective static object group. In a response to a class 1 poll from the master, any event objects that may be present are included for the events of the variables. A master can assign the variables by means of an assign class request, the event class 0, 2 or 3. If the variable is assigned by the master of event class 0, no more events are generated. ▶ two Variables are assigned to the event class 2, and generate events. In a response to a class 0 poll from the master, the static objects are included for the variables. A master can also read the current value of the variables by means of a read request for the respective static object group. In a response to a class 2 poll from the master, any event objects that may be present are included for the events of the variables. A master can assign the variables by means of an assign class request, the event class 0, 1 or 3. If the variable is assigned by the master of event class 0, no more events are generated. ▶ three Variables are assigned to the event class 3, and generate events. In a response to a class 0 poll from the master, the static objects are included for the variables. A master can also read the current value of the variables by means of a read request for the respective static object group. In a response to a class 3 poll from the master, any event objects that may be present are included for the events of the variables. A master can assign the variables by
--	---

	<p>means of an assign class request, the event class 0, 1 or 2. If the variable is assigned by the master of event class 0, no more events are generated.</p> <p>Default: <code>three</code></p>
Generate Analog output events	<p>This option determines whether value changes for variables that are configured for Binary Outputs, events are also to be generated. Not all masters support events for Binary Outputs.</p> <p>Default: <code>active</code></p>

Variables taken from the zenon project are listed with their current configuration in this list.

- ▶ The list can be sorted
 - Click for the sorting on the column heading.
 - The sorting sequence is visualized with an arrow symbol next to the column heading:
 - Arrow upwards: ascending sorting
 - Arrow downwards: descending sorting
 - Another click on the column heading reverses the sorting order.
- ▶ The list can be filtered

To filter the list:

 - Enter the desired filter term in the input field below the heading. The default description of an empty field is `filter text` (shown in gray font).

Parameter	Description
Name	Variable name. The project name is placed in front of the name: ProjectName#VariableName
Identification	Identification of the variable.
Point Index	Current configuration of the DNP3 point index of the variable in this Object Group.
Static Variation	Current configuration of the static variation for the point of the variable in this Object Group.
Event Variation	Current configuration of the event variation for the point of the variable in this Object Group. Note: This parameter is not available in the Octet String node.
Event Class	Current configuration of the event class for the point of the variable in this Object Group.
Control Model	Current configuration of the Control Model . Note: This parameter is only available in the Binary Output node.
Command Routing	Current behavior of the Binary Output point with select and operate requests from the master. Note: This parameter is only available in the Binary Output node.

Supported data types: BOOL, Byte, DINT, DWORD, Date, Date_and_Time, INT, LINT, LREAL, LWORD, REAL, SINT, TIME, TOD, UDINT, UINT, ULINT, USINT, WORD

NAVIGATION BAR

Parameter	Description
Add...	<p>Opens the variable selection dialog (on page 108) to apply the zenon variables in the list.</p> <p>Note: Not active in the Frozen Counter node.</p>
Remove	<p>Removes selected variables from the list.</p> <p>A confirmation dialog is shown before a variable is deleted.</p> <p>Only active if one or more variables in the list are selected.</p> <p>Note: Not active in the Frozen Counter node.</p>
Replace variable...	<p>Opens the variable selection dialog (on page 108) to replace a variable selected in the list with another zenon variable.</p> <p>Only active if precisely one variable in the list is selected.</p>

ANALOG OUTPUT

You configure the parameters for selected variable(s) in the list in this area.

Parameter	Description
Pointer Index	<p>This option determines the DNP3 point index in the Object Group for the selected variable. If variables are added, the point index is automatically set for the newly-added variables and numbered consecutively starting with the highest point index already present.</p> <p>A point index can only be issued once within an Object Group. The entry is validated. If a point index has already been issued, this is shown with a warning dialog.</p> <p>Note: It is recommended that you issue the point index starting with 0 without gaps. The outstation supports the maximum point index. If a higher point index is used, it should be ensured that the master also supports qualifier codes with 2 octets or 4 octets accordingly.</p>
Identification	<p>The identification is primarily taken from the variable from zenon Runtime. The identification can also be amended locally with this option.</p>
Static variation	<p>This option determines the static variation for the selected variable for the point in this Object Group. With the <code>default</code> value, the general static variation defined for this Object Group is used.</p> <p>Note: Select a static variation that is supported by the master. Not every master supports all static variations.</p> <p>Select from drop-down list:</p> <ul style="list-style-type: none"> ▶ 32 Bit with flag Object Group 40 - Variation 1 ▶ 16 Bit with flag Object Group 40 - Variation 2 ▶ Real with flag (Single-precision with flag) Object Group 40 - Variation 3 ▶ LReal with flag (Double-precision with flag) Object Group 40 - Variation 4 <p>Default: <code>default</code></p>
Event variation	<p>This option determines the event variation for the selected variable for the point in this Object Group. With the <code>default</code> value, the general event</p>

	<p>variation defined for this Object Group is used.</p> <p>Note: Select a event variation that is supported by the master. Not every master supports all event variations.</p> <p>Select from drop-down list:</p> <ul style="list-style-type: none"> ▶ 32 Bit without time Object Group 42 - Variation 1 ▶ 16 Bit without time Object Group 42 - Variation 2 ▶ 32 Bit with time Object Group 42 - Variation 3 ▶ 16 Bit with time Object Group 42 - Variation 4 ▶ Real without time (Single-precision without time) Object Group 42 - Variation 5 ▶ LReal without time (Double-precision without time) Object Group 42 - Variation 6 ▶ Real with time (Double-precision with time) Object Group 42 - Variation 7 ▶ LReal with time (Double-precision with time) Object Group 42 - Variation 8 <p>Default: default</p>
Event class	<p>This option determines the event class for the selected variable for the point in this Object Group. With the <code>default</code> value, the general event class defined for this Object Group is used.</p> <p>Selection: see description for Default event class option.</p> <p>Default: default</p>
Apply	Applies the configuration for the selected variable(s).

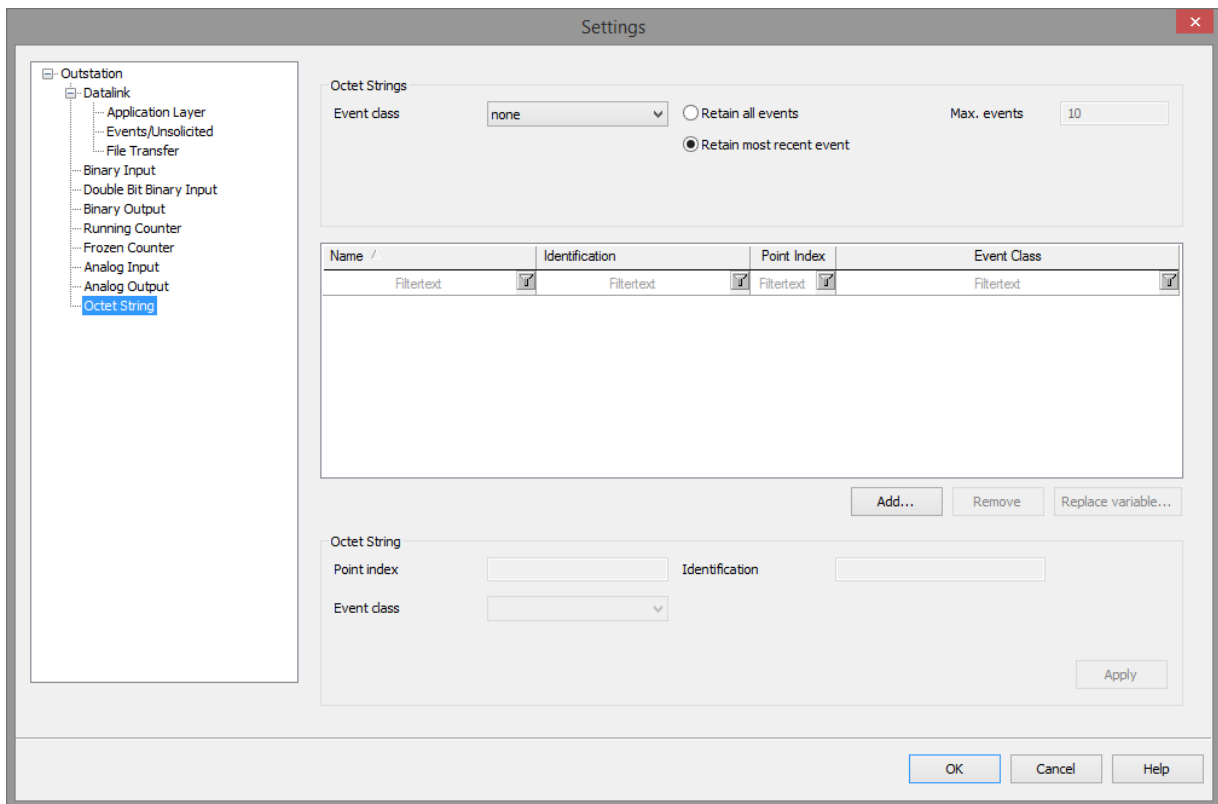
CLOSE DIALOG

Options	Description
OK	Applies settings and closes the dialog.
Cancel	Discards all changes and closes the dialog.

Help

Opens online help.

7.11 Octet String



You configure the DNP3 Octet string points in this node.

Parameter	Description
Event Class	<p>This option determines the default event class for this Object Group for newly-added variables. A different event class can be individually configured for each variable.</p> <p>Select from drop-down list:</p> <ul style="list-style-type: none"> ▶ none <p>Variables are not assigned to an event class and do not generate events. No static objects are included for the variables in a response to a class 0 poll from the master. A master can only read the current value of the variables by means of a read request for the respective static object group. A master can assign the variables by means of an assign class request, not another event class.</p> ▶ zero <p>Variables are assigned to the event class 0, but do not generate any events. In a response to a class 0 poll from the master, the static objects are included for the variables. A master can also read the variables by means of a read request for the respective static object group. A master can assign the variables by means of an assign class request, the event class 1, 2 or 3, as a result of which the variables also generate events from this point in time.</p> ▶ one <p>Variables are assigned to the event class 1, and generate events. In a response to a class 0 poll from the master, the static objects are included for the variables. A master can also read the current value of the variables by means of a read request for the respective static object group. In a response to a class 1 poll from the master, any event objects that may be present are included for the events of the variables. A master can assign the variables by means of an assign class request, the event class 0, 2 or 3. If the variable is assigned by the master of event class 0, no more events are generated.</p> ▶ two <p>Variables are assigned to the event class 2, and generate events. In a response to a class 0 poll from the master, the static objects are included for the variables. A master can also read the current value of the variables by means of a read request for the respective static object group. In a response to a class 2 poll from the master, any event objects that</p>

	<p>may be present are included for the events of the variables. A master can assign the variables by means of an assign class request, the event class 0, 1 or 3. If the variable is assigned by the master of event class 0, no more events are generated.</p> <p>► <code>three</code> Variables are assigned to the event class 3, and generate events. In a response to a class 0 poll from the master, the static objects are included for the variables. A master can also read the current value of the variables by means of a read request for the respective static object group. In a response to a class 3 poll from the master, any event objects that may be present are included for the events of the variables. A master can assign the variables by means of an assign class request, the event class 0, 1 or 2. If the variable is assigned by the master of event class 0, no more events are generated.</p> <p>Default: <code>none</code></p>
Event retention	<p>Behavior when forwarding value changes.</p> <p>Selection from option field.</p> <p>► <code>Retain all events</code> Each time a value, status or time stamp of a variable is changed, a new event is created and saved in the event buffer. The master gets all changes.</p> <p>► <code>Retain most recent event</code> The event buffer buffer only contains one event for the last change of value, status or time stamp. The master only gets the last change and, under certain circumstances, cannot distinguish whether or how often a value has changed in the meantime.</p> <p>Default: <code>Retain most recent event</code></p>
Max. events	<p>Maximum number of events that are saved in the event buffer for this Object Group for all defined variables. If <code>Retain all events</code> is configured, this number should be at least as large as the number of configured variables.</p> <p>Default: 100</p> <p>Note: Only active if <code>Retain all events</code> is active.</p>

Variables taken from the zenon project are listed with their current configuration in this list.

- The list can be sorted

- Click for the sorting on the column heading.
 - The sorting sequence is visualized with an arrow symbol next to the column heading:
 Arrow upwards: ascending sorting
 Arrow downwards: descending sorting
 - Another click on the column heading reverses the sorting order.
- The list can be filtered
 To filter the list:
- Enter the desired filter term in the input field below the heading. The default description of an empty field is `filter text` (shown in gray font).

Parameter	Description
Name	Variable name. The project name is placed in front of the name: <code>ProjectName#VariableName</code>
Identification	Identification of the variable.
Point Index	Current configuration of the DNP3 point index of the variable in this Object Group.
Static Variation	Current configuration of the static variation for the point of the variable in this Object Group.
Event Variation	Current configuration of the event variation for the point of the variable in this Object Group. Note: This parameter is not available in the Octet String node.
Event Class	Current configuration of the event class for the point of the variable in this Object Group.
Control Model	Current configuration of the Control Model . Note: This parameter is only available in the Binary Output node.
Command Routing	Current behavior of the Binary Output point with select and operate requests from the master. Note: This parameter is only available in the Binary Output node.

Supported data types: `STRING`, `WSTRING`

NAVIGATION BAR

Parameter	Description
Add...	<p>Opens the variable selection dialog (on page 108) to apply the zenon variables in the list.</p> <p>Note: Not active in the Frozen Counter node.</p>
Remove	<p>Removes selected variables from the list.</p> <p>A confirmation dialog is shown before a variable is deleted.</p> <p>Only active if one or more variables in the list are selected.</p> <p>Note: Not active in the Frozen Counter node.</p>
Replace variable...	<p>Opens the variable selection dialog (on page 108) to replace a variable selected in the list with another zenon variable.</p> <p>Only active if precisely one variable in the list is selected.</p>

OCTET STRING

You configure the parameters for selected variable(s) in the list in this area.

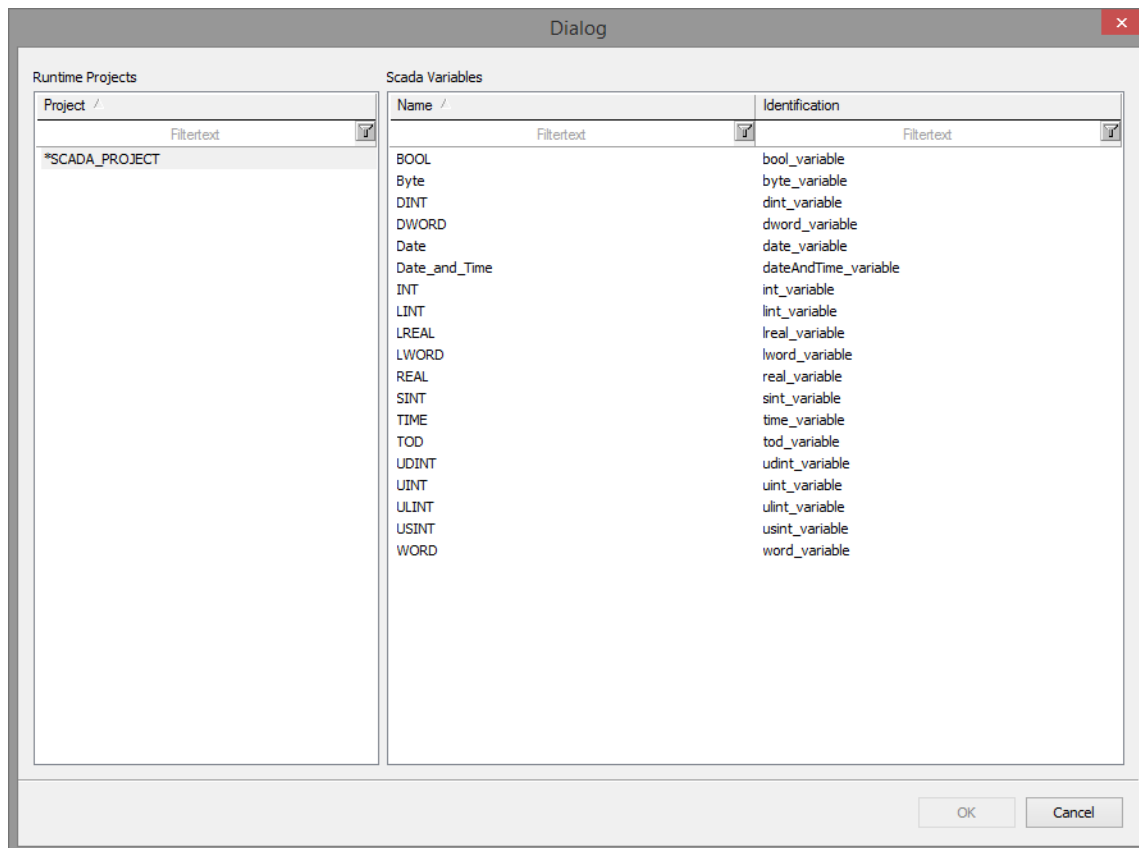
Parameter	Description
Point index	<p>This option determines the DNP3 point index in the Object Group for the selected variable. If variables are added, the point index is automatically set for the newly-added variables and numbered consecutively starting with the highest point index already present.</p> <p>A point index can only be issued once within an Object Group. The entry is validated. If a point index has already been issued, this is shown with a warning dialog.</p> <p>Note: It is recommended that you issue the point index starting with 0 without gaps. The outstation supports the maximum point index. If a higher point index is used, it should be ensured that the master also supports qualifier codes with 2 octets or 4 octets accordingly.</p>
Identification	<p>The identification is primarily taken from the variable from zenon Runtime. The identification can also be amended locally with this option.</p>
Event class	<p>This option determines the event class for the selected variable for the point in this Object Group. With the <code>default</code> value, the general event class defined for this Object Group is used.</p> <p>Selection: see description for Default event class option.</p> <p>Default: <code>default</code></p>
Apply	Applies the configuration for the selected variable(s).

CLOSE DIALOG

Options	Description
OK	Applies settings and closes the dialog.
Cancel	Discards all changes and closes the dialog.
Help	Opens online help.

7.12 Variable selection dialog

In this dialog, you apply variables from an existing zenon project configuration in the **DNP3_SG** configuration. Multiple selection is possible.



- ▶ The list can be sorted
 - Click for the sorting on the column heading.
 - The sorting sequence is visualized with an arrow symbol next to the column heading:
 Arrow upwards: ascending sorting
 Arrow downwards: descending sorting
 - Another click on the column heading reverses the sorting order.
- ▶ The list can be filtered
 To filter the list:
 - Enter the desired filter term in the input field below the heading. The default description of an empty field is `filter text` (shown in gray font).

RUNTIME PROJECTS

Parameter	Description
[Project name]	Name of the zenon projects loaded in Runtime. The active project is marked with a * (star character)

SCADA VARIABLES

Parameter	Description
[Variable list]	<p>List of the configured variables in the currently-loaded zenon projects. The list offers, for selection, the data types that are supported for the respective nodes.</p> <ul style="list-style-type: none"> ▶ Name : Name of the variable. This corresponds to the Name variable property in the current zenon configuration. ▶ Identification : Identification of the variable. This corresponds to the Identification variable property in the current zenon configuration.

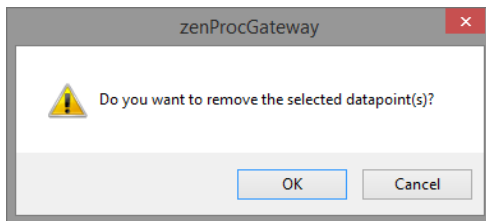
CLOSE DIALOG

OK	<p>Applies settings and closes the dialog.</p> <p>The selected variables are applied in the DNP3_SG configuration of the current node.</p>
Cancel	<p>Discards all changes and closes the dialog.</p> <p>No variables are applied in the DNP3_SG configuration.</p>

7.13 Warning dialog

REMOVE VARIABLE (DATA POINT)

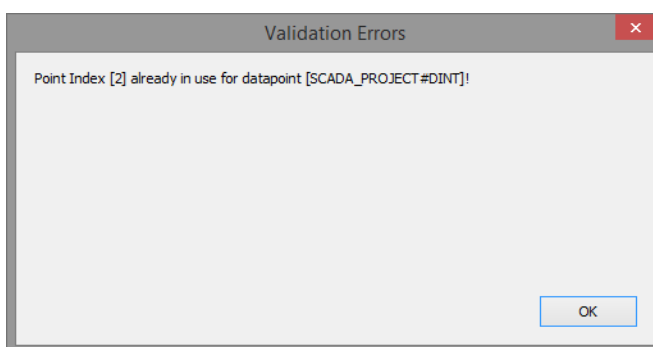
The following warning dialog is shown before a variable is deleted:



Parameter	Description
OK	Deletes the selected variable(s) from the list.
Cancel	Closes the dialog without deleting the selected variable(s).

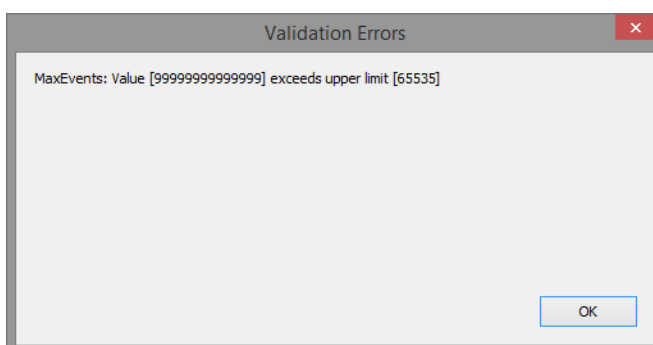
DATA POINT NOT UNIQUE

If a point index that is already present in the current configuration has been configured for a variable, this is shown with a warning dialog.



INVALID INPUT

If an invalid entry has been configured for an option, this is shown in a warning dialog.



7.14 Status mapping zenon - DNP3

The point mapping of the zenon status to the DNP status is carried out if an event is logged in the event buffer.

The mapping between zenon and DNP 3 is carried out as follows:

	Status DNP3							
Status zenon	ONLINE (Note 2)	COMM_LOST (Note 2)	LOCAL_FORCED (Note 2)	OVER_RANGE (Note 3)	REFERENCEERR (Note 3)	DISCONTINUITY (Note 3)	ROLLOVER (Note 3)	CHATTERFILTER (Note 3)
SPONT (Note 1)	1	0	0					
GA (Note 1)	1	0	0					
INVALID (Note 1)	0	1	0					
ALTVAL (Note 1)	0	0	1					
OFF (Note 1)	0	0	0					

N_UPDATE (Note 1)	0	0	0					
NT_870 (Note 1)	0	0	0					
SB_870 (Note 1)	1	0	1					
OV_870				1				
OR_870				1				

NOTES

► Note 1:

At least one status must be set in zenon. A corresponding LOG entry is created if this is not the case. No additional flags are used.

► Note 2:

Forced base flags, these flags are always set as a bit field.

► Note 3:

Flags are only set for ONLINE or LOCAL_FORCED.

7.15 Select before Operate

Analog Outputs and Binary Outputs have SBO (Select before Operate) functionality. For this, the following applies:

- ▶ The DNP3 master sends a `Select` command to the outstation.
- ▶ When the `Binary Output` object type is requested, the following requirements must be met in order for the request to be answered in the positive:
 - The variable exists and is signed in or can be signed in.
 - The data type of the variable is `IEC_BOOL` or `IEC_USINT`
 - The value of the command is valid:
`NUL`, `PulseOn`, `PulseOff`, `LatchOn`, `LatchOff`, `PulseOn + Close`, `PulseOn + Trip`
- ▶ Requests of the `Analog Output` object type are always answered in the positive
- ▶ The configured `Select` timeout is activated automatically.
 If, within this timeout, there is no `Operate` command received for the respective variable, the processing of the variable is ended automatically.
Note: If, instead of an `Operate` Request, a different Request is received for the respective variable, the sequence number is incorrect or the `Operate` Request does not correspond to the `Select` Request. Processing is ended automatically.
- ▶ An `Operate` Request without a previous `Select` Request is acknowledged with an error.



Hint

You can find further general information on *Select before Operate* in the Energy Edition manual, **Command Processing** in the Execution of a command chapter.

7.16 Command processing - Command Routing

Basic requirements for the positive execution of the `Select` or `Operate` command are:

- ▶ The variable type is `Binary Output`
- ▶ The variable exists, is signed in or can be signed in, and is valid (no `I-Bit`).
- ▶ The data type of the variable is `IEC_REAL`, `IEC_BOOL` or `IEC_USINT`.
- ▶ The control code is valid and can be converted to a Boolean value -
 The following values are valid (the corresponding Boolean value is stated in brackets):
 - `NUL` (`false`)

- PulseOn (true)
- PulseOff (false)
- LatchOn (true)
- LatchOff (false)
- PulseOn + Close (true)
- PulseOn + Trip (false)

PROCEDURE

In general, the following applies: Command routing is only activated for exported Binary Output variables for which the **Command Routing** option has been activated.

- ▶ If a `Select` command is received for a Binary Output for which the **Command Routing** option has been deactivated, a positive `Response` is returned to the master, provided the basic requirements have been met.
- ▶ If, for Binary Output, the **Command Routing** option has been set, the `COT_act` command with the control code is sent to the command variable (configured with **Binary Output**), in the Runtime.
Note: `COT_act` is coded in the status value with the help of the `COT_act`, `S_SE_BIT` and `S_SELECT` bits.
- ▶ There is then a wait for the execution of the `Select` command by the auto-remote command action, before the `Response` is sent to the master.
Only once there is a positive response to the `Select` command sent to the master does the `Select Timeout Timer` start.
- ▶ The processing of the following `Operate` command is carried out in the same way as for the `Select` command. In addition, for an `Operate` command, there is a check to see whether a `Select`-command has been received before the command is forwarded to the command processing in Runtime.
- ▶ If, for the `Operate` command, the **Command Routing** option is not active or no `Select` command has been received beforehand, an `Operate Request` is answered directly.
- ▶ After the `Operate` command has been processed successfully by the command processing in Runtime, a positive `Operate Response` is sent to the master. In the event of an error, an `Operate Response` is sent to the master with an error.

STATUS OF COMMAND ROUTING

The following status codes are sent to the master from the outstation:

- ▶ **Status Code 0 - SUCCESS**
The command processing was successful.

- ▶ **Status Code 1 - TIMEOUT**
A timeout has occurred during command processing.
The command processing could not be successfully completed within 2 minutes.
- ▶ **Status Code 2 - NO_SELECT**
The `Select` command has been sent by the command processing with an invalid status.
Invalid statuses are:
 - `S_PN_BIT` (negative confirmation) received
 - The `S_SE_BIT` is no longer set
 - None of the `COT` bits are set
- ▶ **Status Code 6 - HARDWARE_ERROR**
This error occurs under the following circumstances:
 - The variable is not signed in and cannot be signed in.
 - The variable has the **INVALID** bit
 - The value cannot be written or the command cannot be sent.
- ▶ **Status Code 10 - AUTOMATION_INHIBIT**
The `Operate` command has been sent by the command processing with an invalid status.
Invalid statuses are:
 - `S_PN_BIT` (negative confirmation) received
 - The `S_SE_BIT` is no longer set
 - None of the `COT` bits are set
- ▶ **Status Code 12 - OUT_OF_RANGE**
The control code contained in the `Request` is invalid.

7.17 LATCH_ON and LATCH_OFF

If **Command Routing** is inactive the Command Control Code from the controller is written to USINT variables as an 8-bit value via the **Process Gateway**.

Control Code		Control Mode	Variable linked	command routing	Act
0x01	Pulse On	Activation	BOOL	disabled	set
0x01	Pulse On	Activation	USINT	disabled	set
0x01	Pulse On	Complementary latch	n.a.	disabled	ret
0x01	Pulse On	Complementary two output	n.a.	disabled	ret
0x01	Pulse On	Activation	BOOL	enabled	set
0x01	Pulse On	Activation	USINT	enabled	set
0x01	Pulse On	Complementary latch	n.a.	enabled	ret
0x01	Pulse On	Complementary two output	n.a.	enabled	ret
0x03	Latch On	Activation	BOOL	disabled	set
0x03	Latch On	Activation	USINT	disabled	set
0x03	Latch On	Complementary latch	BOOL	disabled	set
0x03	Latch On	Complementary latch	USINT	disabled	set
0x03	Latch On	Complementary two output	BOOL	disabled	ret
0x03	Latch On	Complementary two output	USINT	disabled	set
0x03	Latch On	Activation	BOOL	enabled	set
0x03	Latch On	Activation	USINT	enabled	set
0x03	Latch On	Complementary latch	BOOL	enabled	set
0x03	Latch On	Complementary latch	USINT	enabled	set
0x03	Latch On	Complementary two output	BOOL	enabled	set
0x03	Latch On	Complementary two output	USINT	enabled	set
0x04	Latch Off	Activation	BOOL	disabled	set
0x04	Latch Off	Activation	USINT	disabled	set
0x04	Latch Off	Complementary latch	BOOL	disabled	set
0x04	Latch Off	Complementary latch	USINT	disabled	set
0x04	Latch Off	Complementary two output	BOOL	disabled	ret
0x04	Latch Off	Complementary two output	USINT	disabled	set
0x04	Latch Off	Activation	BOOL	enabled	set
0x04	Latch Off	Activation	USINT	enabled	set
0x04	Latch Off	Complementary latch	BOOL	enabled	set
0x04	Latch Off	Complementary latch	USINT	enabled	set
0x04	Latch Off	Complementary two output	BOOL	enabled	set
0x04	Latch Off	Complementary two output	USINT	enabled	set

0x41	Close	Activation	BOOL	disabled	set
0x41	Close	Activation	USINT	disabled	set
0x41	Close	Complementary latch	BOOL	disabled	set
0x41	Close	Complementary latch	USINT	disabled	set
0x41	Close	Complementary two output	BOOL	disabled	ret
0x41	Close	Complementary two output	USINT	disabled	set
0x41	Close	Activation	BOOL	enabled	set
0x41	Close	Activation	USINT	enabled	set
0x41	Close	Complementary latch	BOOL	enabled	set
0x41	Close	Complementary latch	USINT	enabled	set
0x41	Close	Complementary two output	BOOL	enabled	set
0x41	Close	Complementary two output	USINT	enabled	set
0x81	Trip	Activation	BOOL	disabled	set
0x81	Trip	Activation	USINT	disabled	set
0x81	Trip	Complementary latch	BOOL	disabled	set
0x81	Trip	Complementary latch	USINT	disabled	set
0x81	Trip	Complementary two output	BOOL	disabled	ret
0x81	Trip	Complementary two output	USINT	disabled	set
0x81	Trip	Activation	BOOL	enabled	set
0x81	Trip	Activation	USINT	enabled	set
0x81	Trip	Complementary latch	BOOL	enabled	set
0x81	Trip	Complementary latch	USINT	enabled	set
0x81	Trip	Complementary two output	BOOL	enabled	set
0x81	Trip	Complementary two output	USINT	enabled	set

8. DNP3 Slave - AccessDNP3 outdated

From zenon version 8.00, the **DNP3 Slave (AccessDNP3)** was replaced with the new **Process Gateway** module **DNP3_SG**.

- ▶ The outdated **AccessDNP3** module can no longer be selected for new configurations.
- ▶ Existing configurations with the previous **AccessDNP3** module continue to be able to run.

The **AccessDNP3** module will no longer be supplied in a future version. The old module will be completely replaced by the **AccessDNP3_SG** module.



Information

The documentation for the outdated module is no longer supplied. Please contact your support partner at the email address support@copadata.com to request this documentation.

9. ICCP-TASE.2

The **IEC 60870-6 TASE.2 ICCP** protocol can be used with the **AccessICCP.dll** module via the zenon **Process Gateway**. ICCP-TASE.2 supports Conformance Blocks 1 and 2 as client and server. **Process Gateway** acts like a server by default when communication is established.

Communication in the ICCP protocol is spontaneous. An ICCP server reports the value changes to its communication partner, the remote client. However, an ICCP server cannot actively read the value changes from the remote client. In order to guarantee two-way exchange of data, both communication partners should have a client and a server.

POSSIBLE SYSTEM CONFIGURATIONS

► Server only

Communication with several remote ICCP communication partners possible.
This is the case if no variables have been configured on the **Client Variables** tab.

► Server and passive client

The client will not initialize the communication with the remote ICCP server itself.
Communication with several remote ICCP communication partners is possible.
This is the case if the **Initialise communication** property is not active.

Note: **General** tab in the configuration dialog of the **Process Gateway**.

► Server and active client

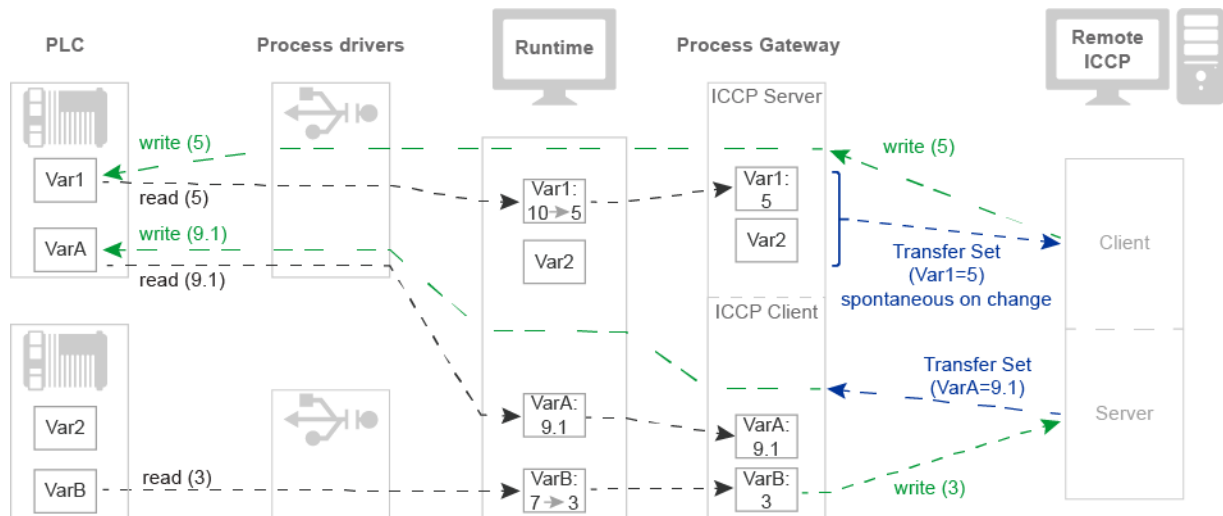
The client can initialize communication with remote ICCP servers itself.
Communication is only possible with a remote ICCP communication partner.

ESTABLISHING COMMUNICATION

After the initialization of the communication with the zenon ICCP server or initialization by its own client, the verification of the **bilateral table ID** is carried out. If the verification was successful, the zenon

ICCP client registers the variables to be communicated on the remote server as a DataSet and activates spontaneous communication of the DSTransferSet (MMS.InformationReport).

Note: The ICCP client also communicates with the server if, in the configuration of the **Client variables** (on page 135) tab, no variables are contained in the **Available from ICCP** list. In this case, the client only checks the bilateral table.



ICCP DATA EXCHANGE

► ICCP Server

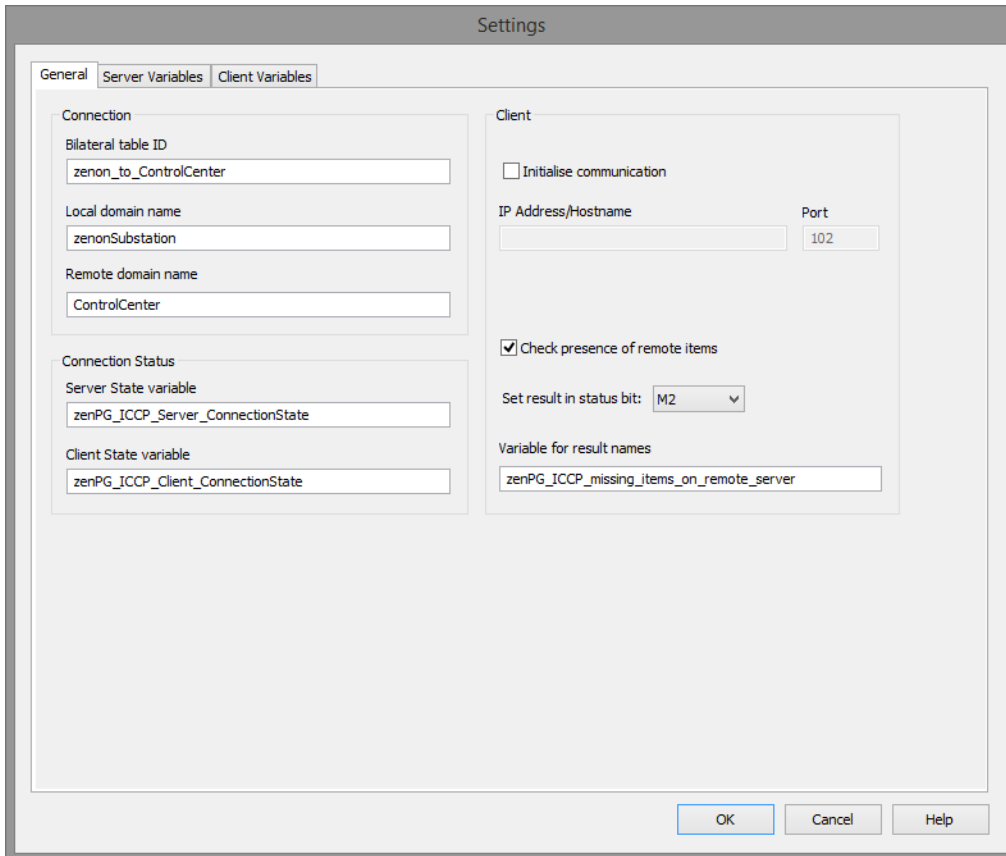
- The ICCP server sends value changes of variables via DSTransferSets (MMS.InformationReport) spontaneously. If the values of variables in zenon Runtime change, a DSTransferSet is sent to the remote client.
- The ICCP server can also receive write requests from the remote client (MMS.write request). These are transferred to zenon Runtime and Runtime forwards this request to the corresponding zenon driver. If the writing is confirmed by the PLC, the zenon variable changes its value.

► ICCP Client

- In the event that a variable in zenon Runtime sends the ICCP client a corresponding write request (MMS.write request) to the remote server.
- If a DSTransferSet (MMS.InformationReport) is received by the ICCP client (from the remote server) the received values are sent to zenon Runtime. Runtime forwards these requests to the zenon driver. If the writing is confirmed by the PLC, the zenon variables do not change their value.

9.1 Module configuration

Configuration is carried out using three tabs. It can be confirmed and closed by clicking on OK if all three tabs have been configured.



The image shows a 'Settings' dialog box with three tabs: 'General', 'Server Variables', and 'Client Variables'. The 'General' tab is active. It contains two main sections: 'Connection' and 'Client'.

Connection Section:

- Bilateral table ID:** zenon_to_ControlCenter
- Local domain name:** zenonSubstation
- Remote domain name:** ControlCenter
- Connection Status:**
 - Server State variable:** zenPG_ICCP_Server_ConnectionState
 - Client State variable:** zenPG_ICCP_Client_ConnectionState

Client Section:

- ☐ Initialise communication
- IP Address/Hostname:** (empty text box)
- Port:** 102
- ☒ Check presence of remote items
- Set result in status bit:** M2 (dropdown menu)
- Variable for result names:** zenPG_ICCP_missing_items_on_remote_server

At the bottom right, there are three buttons: 'OK', 'Cancel', and 'Help'.

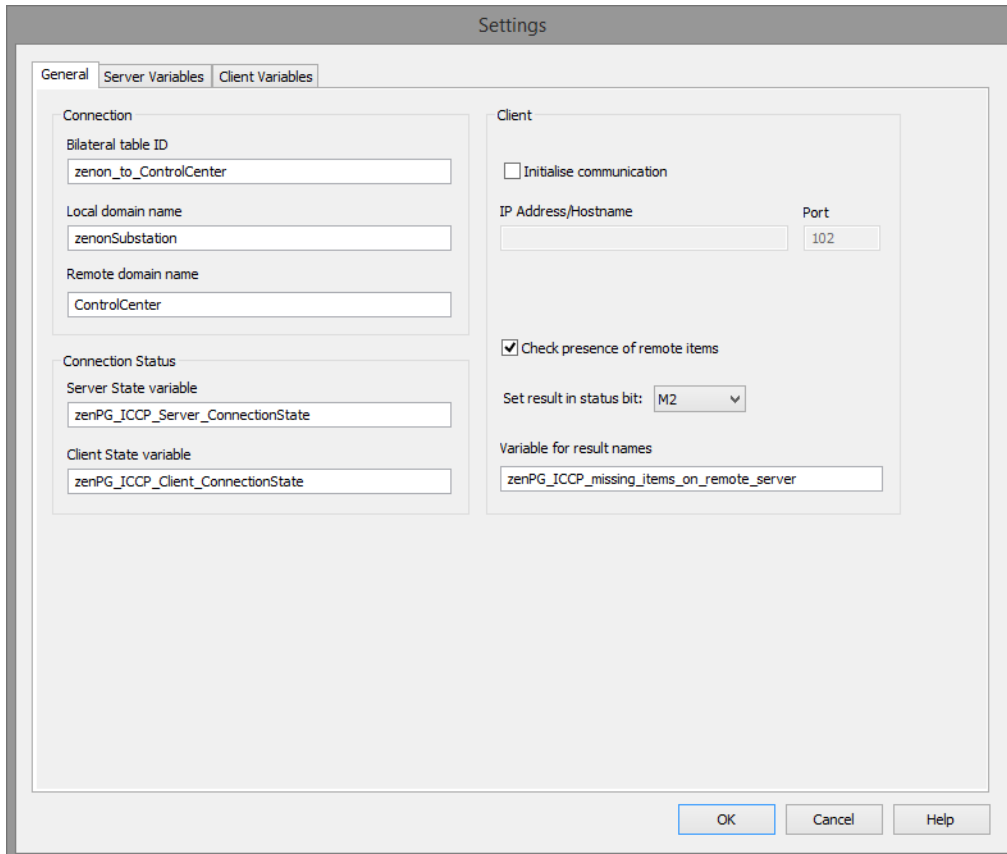
Parameter	Description
General (on page 124)	General settings.
Server variables (on page 131)	Configuration of the server variables. Selection of the variables that the ICCP server provides.
Client variables (on page 135)	Configuration of the client variables. Selection of the variables that are received by the ICCP client.

CLOSE DIALOG

Options	Description
OK	Applies settings and closes the dialog.
Cancel	Discards all changes and closes the dialog.
Help	Opens online help.

9.1.1 General

Configuration of the connection settings.



CONNECTION

The settings in the `Connection` section are any desired strings that are coordinated on all devices in the system.

The unique bilateral table ID must be the same on all devices that communicate with the **ICCP Process Gateway**. It is queried when a connection is established by both communication partners. The connection is disconnected if they do not correspond.

The domain names provide the references of the ICCP objects used at MMS protocol level. It must be set as the same mirror image for the communication partners.

These comparison rules are prescribed by the ICCP standard.

Parameter	Description
Bilateral Table ID	<p>The bilateral table identification is synced when the communication partner establishes a connection. Access to the data is only permitted for authorized remote clients.</p> <p>The comparison is Domain-specific, the communication partner must, in the domain with the name Remote Domain Name, feature the table with the label Bilateral Table ID.</p> <p>Note: The ID must be configured as exactly the same on both sides of the communication.</p> <p>Note: Missing or incorrect entries are visualized as an error message in the dialog.</p>
Local Domain Name	<p>Name of the local ICCP domain that is used for objects of the Process Gateway.</p> <p>Note: the name must be set inversely for the communication partners - the remote client should use the name in its requests to ICCP Process Gateway.</p> <p>Note: Missing or incorrect entries are visualized as an error message in the dialog.</p>
Remote Domain Name	<p>Name of the domains on the remote ICCP server.</p> <p>The Process Gateway ICCP client uses Domain-specific requests with this name:</p> <ul style="list-style-type: none"> ▶ In the comparison with the bilateral table; ▶ For the Data Set and TransferSet on the remote server. Note: a Data Set is created as Domain-specific, its elements - the ICCP data points - are defined as VMD-specific. ▶ In commands via Device objects. <p>Note: Missing or incorrect entries are visualized as an error message in the dialog.</p>
Error message	Display of missing configuration data.

CONNECTION STATUS

Connection status variables allow the display of the current connection status of **Process Gateway** to the remote ICCP communication partner in Runtime. The zenon variable names are given in the **Process Gateway**. The variables should be a numeric data type; recommendation: Internal variables.

You can find detailed information on this in the Status variables (on page 140) chapter.

Parameter	Description
Server state variable	<p>Input field for the entry of a variable name for the display of the current connection status of the server.</p> <p>If this field is empty, no status information can be displayed in zenon Runtime.</p> <p>Default: empty</p>
Client state variable	<p>Input field for the entry of a variable name for the display of the current connection status of the client.</p> <p>If this field is empty, no status information can be displayed in zenon Runtime.</p> <p>Default: empty</p>

CLIENT

Parameter	Description
Initialise communication	<p>Checkbox for the activation of the automatic establishment of a connection to the remote ICCP server when the Process Gateway is started.</p> <ul style="list-style-type: none"> ▶ active: When Process Gateways is started, the client automatically attempts to establish a connection to the configured remote ICCP server. If the connection attempt is unsuccessful, another attempt to establish a connection is made after 20 seconds. During this waiting time, a connection attempt can be accepted from a remote ICCP communication partner if this corresponds to the configured IP address or host name. ▶ inactive: No automatic attempt to establish a connection to the ICCP server when the Process Gateway is started. <p>Default: Inactive</p> <p>If this option is activated, the following limitation applies:</p> <ul style="list-style-type: none"> ▶ The connection is only possible with a remote ICCP communication partner. ▶ Connection attempts from remote clients with an unknown address are not accepted.
IP Address/Hostname	<p>Input field for IP address or host name for the automatic connection to the remote ICCP server.</p> <p>Note: Only active if Initialise communication is inactive.</p>

	Note: Missing or incorrect entries are visualized as an error message in the dialog.
Port	<p>Input field for port number that is used for communication to the remote ICCP server.</p> <p>Default: 102</p> <p>Note: Only active if Initialise communication is inactive.</p> <p>Note: Missing or incorrect entries are visualized as an error message in the dialog.</p>

Check presence of remote items	<p>Checkbox for the activation of the checking of the variable names of the current client connection to the variable name of the remote ICCP server. The names are checked VMD-specifically.</p> <ul style="list-style-type: none"> ▶ active: Activates the checking and allows the configuration of the Set result in status bit and Variable for result names properties. For the connection, the client on the remote server reads a DataSet that only contains existing variables. ▶ inactive: It is assumed that the configured ICCP variables are available on the remote server. If one or more configured ICCP variables are not on the remote server, no successful connection to the remote server can be established. <p>Default: Inactive</p>
Set result in status bit	<p>Status bit for zenon variables of the client that are not on the remote ICCP server. Selection of the corresponding user status bit from drop-down list:</p> <ul style="list-style-type: none"> ▶ None No user status bit is set for the variable if the variable is not present on the remote server. ICCP variables that are not present can be evaluated with Variable for result names. ▶ M1 bis M8 The configured status bit is set for the variable if the variable is not present on the remote server. <p>Default: None</p> <p>This user status bit is set for configured zenon variables if they are not found during checking on the remote server.</p> <p>Note: Only active if Check presence of remote items is inactive.</p> <p>You can find further information on status processing in the Status processing manual in the User status chapter.</p>
Variable for result names	<p>Input field to configure a variable name for the display of the names of the missing ICCP variables. In the variable, variable names that are not present on the remote server are visualized.</p> <p>Is this field is empty, the evaluation is carried out with the configured user status bit.</p> <p>Default: empty</p>

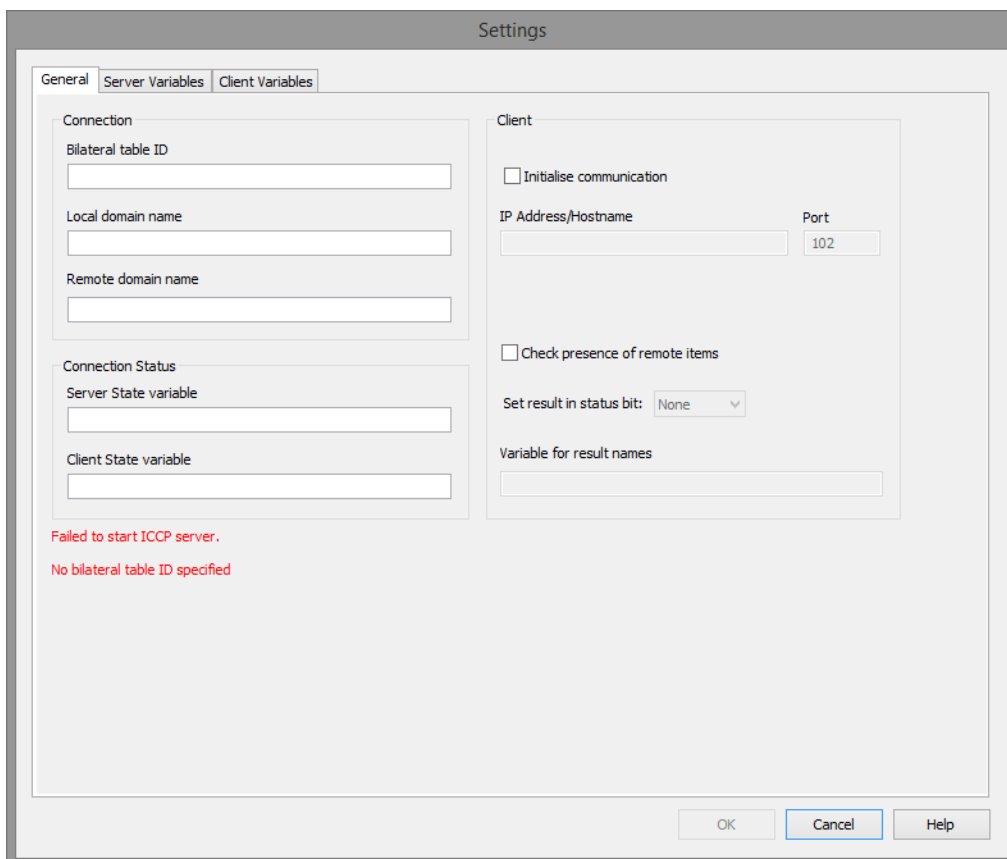
	<p>Note: Only active if Check presence of remote Items is inactive.</p> <p>You can find further information on the display of variables that do not exist in the Show variables that are not present in Runtime (on page 141) chapter.</p>
--	---

CLOSE DIALOG

Parameter	Description
OK	Accepts the settings of all tabs, saves the configuration in the zenProcGateway.ini file, starts the ICCP server and waits for incoming connections.
Cancel	Discards all changes in all tabs and closes the dialog.
Help	Opens online help.

ERROR MESSAGES - EXAMPLE

Error messages are displayed in red font:

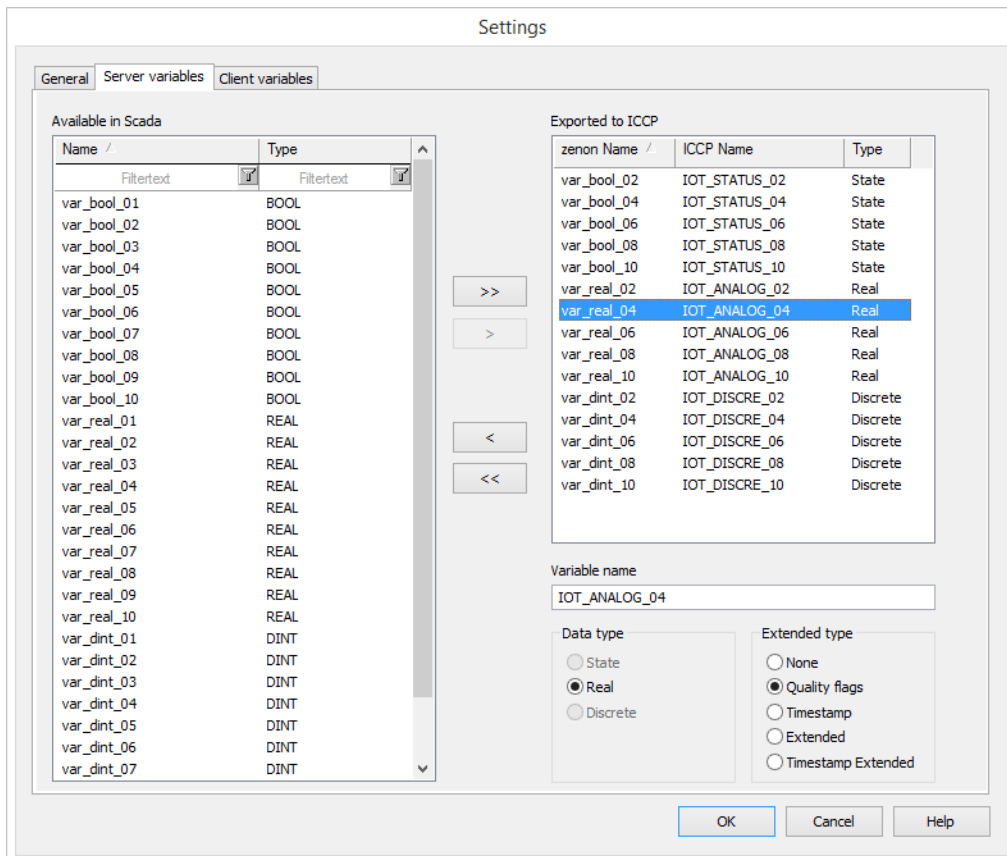


The screenshot shows the 'Settings' dialog box with the 'General' tab selected. The dialog is divided into two main sections: 'Connection' and 'Client'. The 'Connection' section contains fields for 'Bilateral table ID', 'Local domain name', 'Remote domain name', 'Server State variable', and 'Client State variable'. The 'Client' section contains checkboxes for 'Initialise communication' and 'Check presence of remote items', a dropdown for 'Set result in status bit' (currently set to 'None'), and a text field for 'Variable for result names'. At the bottom of the dialog, there are 'OK', 'Cancel', and 'Help' buttons. Two error messages are displayed in red font at the bottom left of the dialog area:

- Failed to start ICCP server.
- No bilateral table ID specified

9.1.2 Server variables

Selection and configuration of the variables that the zenon Process Gateway provides as an ICCP server. The value changes of the variables are sent to the remote ICCP client spontaneously.



Parameter	Description
SCADA variables	<p>List of all numeric variables available in the zenon project. The list can be sorted and filtered.</p> <p>The list shows:</p> <ul style="list-style-type: none"> ▶ Name: Name in zenon ▶ Type: Data type of the zenon variable
Arrow keys	<p>Selected variables are exchanged between the two lists by means of the arrow key.</p> <ul style="list-style-type: none"> ▶ Key >>: Copies all variables from zenon to ICCP. ▶ Key <<: Removes all variables from the ICCP list. ▶ Key >: Copies selected variables from zenon to ICCP ▶ Key <: Removes selected variables from the ICCP list. <p>Variables can also be added to the list for zenon by double clicking on the variable for the ICCP list.</p>
Available for ICCP	<p>Variables from the list are provided by the server for the remote ICCP client.</p> <p>The list shows:</p> <ul style="list-style-type: none"> ▶ SCADA Variable: Name in zenon ▶ ICCP Name: Name in the ICCP protocol ▶ Type: ICCP Type <p>The ICCP name and type can be modified using the ICCP Name, Data type and Extended type options.</p>
ICCP Name	<p>Name - VMD-specific - of the ICCP data point selected from the top list.</p> <p>Change to the ICCP Name by entering the new name. Empty if no entry is selected in the Available for ICCP list.</p> <p>Note: The ICCP name can usually be named as desired in accordance with the standard. However, it must be ensured that both communication partners use the same name.</p>
Data type	<p>ICCP data type of the variable selected from the top list.</p> <p>Change to the Data type by clicking on the option field:</p>

- ▶ **State**
Not available if, in the **Object type** option, the `Device` object option field has been selected.
- ▶ **Command**
This option field is only available if, in the **Object type** option field, the `Device` object option field has been activated.
- ▶ **Real**
- ▶ **Discrete**

When transferring the list from the list of **SCADA Variables**, the `Type` is automatically proposed for the **Available from ICCP** list.

Note: Standard display of zenon variables on ICCP:

- ▶ `USINT/BOOL` to **State**; value range 0..3, **DPI mapping**:
 - 0/false - off (tripped),
 - 1/true - on (closed),
 - 2 - intermediate (between),
 - 3 - fault (invalid).
- ▶ `REAL` to **Real**.
- ▶ `DINT/INT/SINT` to **Discrete**.

Depending on the variables provided by zenon, data types can also be unavailable (grayed out).

Extended type	<p>ICCP suffix of the variable selected from the top list.</p> <p>Change to the Extended type by clicking on the option field:</p> <ul style="list-style-type: none"> ▶ None: No ICCP addition (only value of the variable); IEC 60870-6-802 IndicationPoint type: Data_Discrete Data_State, Data_Real. ▶ Quality flags: Value with quality; for example: Data_DiscreteQ. ▶ Timestamp: Value + quality + time stamp [s]; for example: Data_DiscreteQTimeTag. ▶ Extended: Value + quality + time stamp [s] + change counter; for example: Data_DiscreteExtended. ▶ Timestamp Extended: Value + quality + extended time stamp [ms]; for example: Data_DiscreteQTimeTagExtended. <p>Depending on the variables provided by zenon, data types can also be unavailable (grayed out).</p>
----------------------	---



Attention

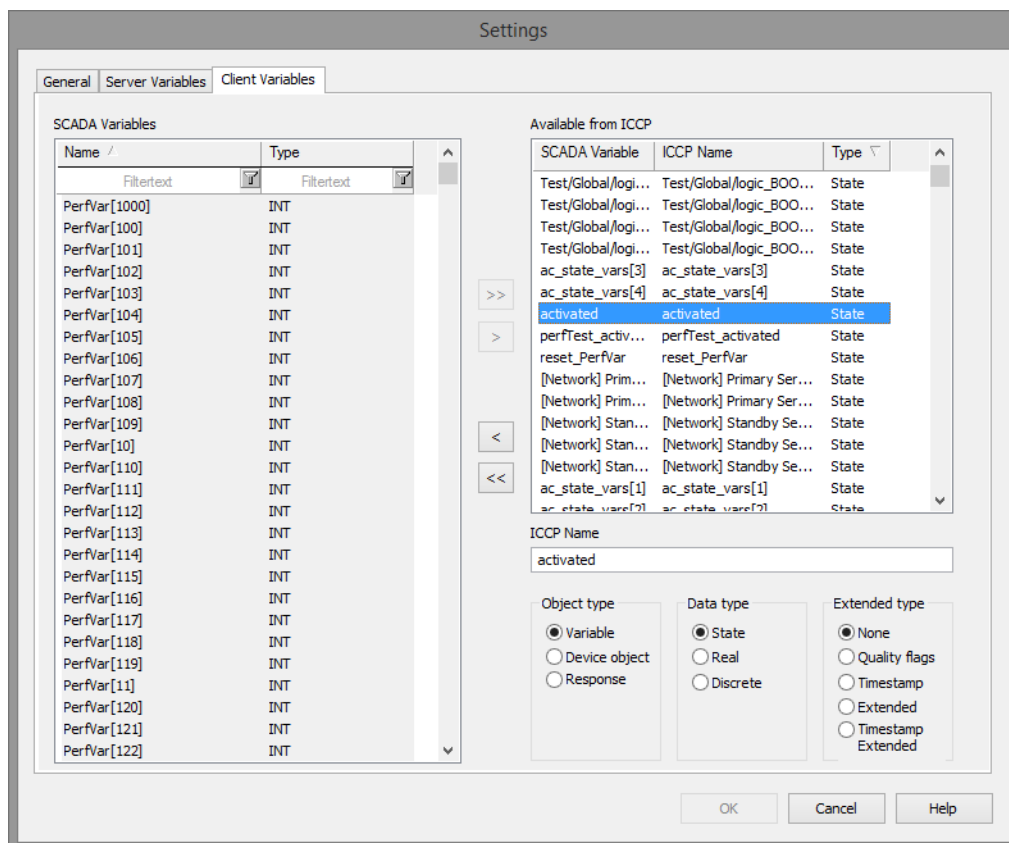
Data type and **Extended type** must be correctly assigned for both communication partners. Incorrect amendments lead to the received data sets not being interpreted 100% correctly. All other information in the data set is lost!

CLOSE DIALOG

Parameter	Description
OK	Accepts the settings of all tabs, saves the configuration in the zenProcGateway.ini file, starts the ICCP server and waits for incoming connections.
Cancel	Discards all changes in all tabs and closes the dialog.
Help	Opens online help.

9.1.3 Client variables

Selection and configuration of the variables that the zenon Process Gateway uses as an ICCP client. The client transmits the value changes that it receives from the remote ICCP server to the zenon variables.



Parameter

SCADA variables

Description

List of all numeric variables available in the zenon project. The list can be sorted and filtered.

The list shows:

- ▶ Name:
Name in zenon
- ▶ Type:
Data type of the zenon variable

Arrow keys

Selected variables are exchanged between the two lists by means of the arrow key.

- ▶ Key >>:
Copies all variables from zenon to ICCP.
- ▶ Key <<:
Removes all variables from the ICCP list.
- ▶ Key >:
Copies selected variables from zenon to ICCP
- ▶ Key <:
Removes selected variables from the ICCP list.

Variables can also be added to the list for zenon by double clicking on the variable for the ICCP list.

Available from ICCP

Variables from the list are expected from the remote ICCP server.

The list shows:

- ▶ SCADA Variable:
Name in zenon
- ▶ ICCP Name:
Name in the ICCP protocol
- ▶ Type:
ICCP Type

The ICCP name and type can be modified using the **ICCP Name**, **Data type** and **Extended type** settings.

ICCP Name

Name - VMD-specific - of the ICCP data point selected from the top list.

Change to the **ICCP Name** by entering the new name.

Empty if no entry is selected in the **Available for ICCP** list.

Note: The ICCP name can usually be named as desired in accordance with the standard. However, it must be ensured that both communication partners use the same name.

Parameter**Object type****Description**

Type of the SCADA variable for communication to the ICCP server. With this object, variables can be configured as command or response variables. Change the `Object type` by clicking on the option field:

- ▶ **Variable**
Standard type of a variable.
Activate this option if the zenon variable is not a command variable or a response variable.
- ▶ **Device object**
The variables from the list of **SCADA Variables** are configured as a command variable if the option is activated. In doing so, the Process Gateway, on receipt of a spontaneous value, sends an `Operate Request` to the ICCP server.
By activating this option field, in the **Data type** option, the `State` option field switches the selection possibility to `Command`.
- ▶ **Response**
The variable from the **SCADA Variables** list is configured as a response variable if the option is activated. When receiving a spontaneous value, the response from the server (`Indication`) is written to the response variable via the `operate` command. No further project configuration is necessary for this process.

Default: Variable

Parameter

Data type

Description

ICCP data type of the variable selected from the top list.

Change to the `Data type` by clicking on the option field:

- ▶ **State**
Not available if, in the **Object type** option, the `Device object` option field has been selected.
- ▶ **Command**
This option field is only available if, in the **Object type** option field, the `Device object` option field has been activated.
- ▶ **Real**
- ▶ **Discrete**

When transferring the list from the list of **SCADA Variables**, the `Type` is automatically proposed for the **Available from ICCP** list.

Note: Standard display of zenon variables on ICCP:

- ▶ `USINT/BOOL` to **State**; value range 0..3, **DPI mapping**:
 - 0/false - off (tripped),
 - 1/true - on (closed),
 - 2 - intermediate (between),
 - 3 - fault (invalid).
- ▶ `REAL` to **Real**.
- ▶ `DINT/INT/SINT` to **Discrete**.

Depending on the variables provided by zenon, data types can also be unavailable (grayed out).

Note: Hidden if the `Response` action field is activated for the **Object Type** option.

Parameter

Extended type

Description

ICCP suffix of the variable selected from the top list.

Change to the **Extended type** by clicking on the option field:

- ▶ None:
No ICCP addition (only value of the variable);
IEC 60870-6-802 IndicationPoint type: Data_Discrete Data_State, Data_Real.
- ▶ Quality flags:
Value with quality;
for example: Data_DiscreteQ.
- ▶ Timestamp:
Value + quality + time stamp [s];
for example: Data_DiscreteQTimeTag.
- ▶ Extended:
Value + quality + time stamp [s] + change counter;
for example: Data_DiscreteExtended.
- ▶ Timestamp Extended:
Value + quality + extended time stamp [ms];
for example: Data_DiscreteQTimeTagExtended.

Depending on the variables provided by zenon, data types can also be unavailable (grayed out).

Note: Hidden if either the `Device object` or `Response option` field is activated for the **Object Type** option.



Attention

Data type and **Extended type** must be correctly assigned for both communication partners. Incorrect amendments lead to the received data sets not being interpreted 100% correctly. All other information in the data set is lost!

CLOSE DIALOG

Parameter	Description
OK	Accepts the settings of all tabs, saves the configuration in the zenProcGateway.ini file, starts the ICCP server and waits for incoming connections.
Cancel	Discards all changes in all tabs and closes the dialog.
Help	Opens online help.

9.2 Status variables of the connection

Status variables of communication are used for the display of the connection status between the Process Gateway and the remote ICCP communication partner. A separate variable each is configured for the ICCP server and ICCP client. The Process Gateway only communicates this status to zenon Runtime. There is no communication of values of these variables to remote ICCP communication partners.

ENGINEERING

To show the connection status in Runtime, carry out the following project configuration steps:

1. In your zenon Editor, create two numeric variables.
It is recommended that these are configured for the internal driver.
 - The variable cannot be a STRING variable.
2. Configuration in the Process Gateway:
 - Status variable for server connection:
Enter the name of the variable in the **Server state variable** input field in the **General** tab in the Process Gateway module configuration.
 - Status variable for client connection:
Enter the name of the variable in the **Client state variable** input field in the **General** tab in the Process Gateway module configuration.

ASSIGNED STATES

The connection status variables have the following values:

Parameter	Description
0 IDLE	<ul style="list-style-type: none"> ▶ Communication inactive: Directly after the start of Process Gateway or reloading the zenon project. ▶ Connection disconnected again: Once Process Gateway has been closed or after the settings in Process Gateway have been edited, and before the establishing of a new connection.
1 CONNECTING	<p>Communication via TCP/IP and MMS is being established.</p> <p>This status is shown if an incoming CR (Connection Request) has been received or a CR (Connection Request) has been sent successfully. This depends on which page has established the TCP/IP connection. This status is the same for server and client.</p>
2 CONNECTED	<p>ICCP communication has been established successfully.</p> <p>This status is set to a logical connection if a dataset (DSTransferSet) has been agreed for the transfer of data and the first MMS.write request (logical server connection) or the first MMS.write response (logical client connection) has been received.</p> <p>If the ICCP client does not request any data from the remote server, the status of the logical connection remains CONNECTED.</p> <p>Note: The client does not request any data if:</p> <ul style="list-style-type: none"> ▶ The list of variables is empty Available from ICCP. ▶ None of the variables from the Available from ICCP list are available and the Check presence of remote items check has been activated.
3 ERROR	<p>This status is shown if the TCP/IP connection has been interrupted. This status is the same for both logical connections.</p>

9.3 Show non-present variables in Runtime

With the help of this project configuration, you can evaluate in zenon Runtime which of the variables configured on the ICCP client are not present on the remote server.

This evaluation can be carried out with two project configurations:

1. Display of the missing variable names in an additional string variable.
2. Display of a user status bit when there are variables that do not exist.

REQUIREMENT

- ▶ The checking of the variable names of the current client connection must be activated: Checkbox **Check presence of remote items** in the **General** tab in the Process Gateway module configuration.

DISPLAY OF THE MISSING VARIABLES IN A STRING VARIABLE

To display the missing variables in Runtime in one variable:

1. In the zenon Editor, create a STRING variable.
It is recommended that these are configured for the internal driver.
2. Configuration in Process Gateway:
Enter the name of the variable in the input field **Variable for result names** in the **General** tab in the Process Gateway module configuration.



Information

The value of the variable in Runtime contains the name of the non-present variables, separated by a comma (;).

DISPLAY OF A USER STATUS BIT WHEN THERE ARE VARIABLES THAT DO NOT EXIST.

To mark the zenon variables in Runtime with a user status bit if they are not present on a remote ICCP server:

- ▶ Select the desired user status bit from the drop-down list of the **Set result in status bit** property in the **General** tab in the Process Gateway module configuration.

Note: This user status bit can be evaluated in the Runtime with a **reaction matrix** or a **combined element**.

9.4 Direct control (non-SBO) for ICCP control objects

- ▶ A zenon variable can be linked to an ICCP control object. A **Direct Control** is executed on receipt of a spontaneous value for this command variable. In doing so, the process gateway sends an **Operate Request** to the ICCP server.
- ▶ A zenon variable can be linked to an ICCP control object as a response variable. The result of the **Direct Control** operation (= indication) is written to the configured zenon result variable (by writing a set value).

- ▶ Each received `indication` is written to the corresponding ICCP response variable.
 - In doing so, there is no check to see whether a command is pending.
 - If no `indication` is received, the value of the response variable remains unchanged.
- ▶ `direct control - supported telegram types`
 - `Operate (request)`
 - `Timeout (indication)`
 - `Local Reset (indication)`
 - `Success (indication)`
 - `Failure (indication)`
- ▶ The response variable is set to a stipulated value when sending a new command.
- ▶ The configuration is validated when the gateway is initialized:
 - If several response variables are configured for each `device object`, an error message is created when the **Process Gateway** is started and only the first one is used
 - Only one command can be configured per `device object`.
 - A command variable must be present for each response variable, otherwise there will be an error message.

PROJECT CONFIGURATION IN THE PROCESS GATEWAY

- ▶ Open the tab in the **Process Gateway. Client Variables**
- ▶ Configure command variable
- ▶ Configure response variable

A response from an ICCP server is assigned to a **Device object** or to a variable by getting the same ICCP name.

RESULT OF THE WRITE REQUEST

If a **Response Object** is configured to a variable or a **Device Object**, the result of the `write Requests` is written to this object:

Parameter	Description
0	Write successful
1	Write pending - waiting for response
200	Write failed: OBJECT_INVALIDATED
201	Write failed: HW_FAULT
202	Write failed: TEMPORARILY_UNAVAILABLE
203	Write failed: OBJECT_ACCESS_DENIED
204	Write failed: OBJECT_UNDEFINED
205	Write failed: INVALID_ADDRESS
206	Write failed: TYPE_UNSUPPORTED
207	Write failed: TYPE_INCONSISTENT
208	Write failed: OBJECT_ATTRIBUTE_INCONSISTENT
209	Write failed: OBJECT_ACCESS_UNSUPPORTED
210	Write failed: OBJECT_NON_EXISTENT
211	Write failed: OBJECT_VALUE_INVALID

9.5 Configuration file: INI entries for ICCP-Tase.2

The configuration file `zenProcGateway.ini` contains this entries for ICCP-Tase.2.

[ICCP]

Entry	Description
SERVER_PORT=	<p>Number of the IP port of the server; the port allows the connection of remote clients.</p> <p>The port is opened on startup and remains open.</p> <p>Default: 102</p> <p>Note: This INI entry cannot be configured in the graphical user interface for the ICCP-TASE.2 Process gateway.</p>
MAX_CONNECTIONS=	<p>Maximum number of connections of remote ICCP clients to the server. Is ignored if CLIENT_AUTO_CONNECT=1 has been configured.</p> <p>Default: 0 (= unlimited)</p> <p>Note: This INI entry cannot be configured in the graphical user interface for the ICCP-TASE.2 Process gateway.</p>
MAJOR_VERSION_NUMBER=	<p>Main version number of the ICCP protocol, for example: 1996, 2000.</p> <p>Default: 2000</p> <p>Note: This INI entry cannot be configured in the graphical user interface for the ICCP-TASE.2 Process gateway.</p>
MINOR_VERSION_NUMBER=	<p>Sub-version number of the ICCP protocol.</p> <p>Default: 8</p> <p>Note: This INI entry cannot be configured in the graphical user interface for the ICCP-TASE.2 Process gateway.</p>
BILATERAL_TABLE_ID=	<p>Bilateral table identification. This is synchronized when establishing a connection between all ICCP communication partners.</p> <p>Note: The unique bilateral table ID must be the same on all devices that communicate with the ICCP-TASE.2 Process Gateway.</p>
LOCAL_DOMAIN_NAME=	<p>Local ICCP domain name; setting for server, should be set as mirror-inverted to the communication partner.</p>
REMOTE_DOMAIN_NAME=	<p>Local ICCP domain name; setting for client, should be set as mirror-inverted to the communication partner.</p>
CLIENT_AUTO_CONNECT=	<p>Project configuration of whether the client automatically initiates the establishment of a connection to the remote ICCP server when Process Gateway is started.</p> <p>0: do not initialize - just wait</p> <p>1: initialize</p>

CLIENT_IP_NAME=	IP address or host name of the remote server if this client is to initialize communication.
CLIENT_PORT=	Port number of the remote server if this client is to initialize communication.
CHECK_SERVER_Variables=	Check to see whether all configured variables are available on the remote server. 0: no check of configured variable names. 1: Check of configured variable names.
CHECK_SERVER_VARIABLES_MARKER_BIT=	Status bit for client variables that are not present on the server if a check of configured variables is activated. 0: no status bit for missing client variables 1 to 8: User status bit M1 to M8
SERVER_STATE_VARIABLE=	Name of the variable for the display of the connection status between ICCP server and remote client.
CLIENT_STATE_VARIABLE=	Name of the variable for the display of the connection status between ICCP client and remote server.
NOT_AVAILABLE_SERVER_VARIABLES_VARIABLE=	Name of the variable for the display of variable names (in zenon Runtime) that are not available for the current client connection on the remote server.
CALLING_OSI_SESSION_SELECTOR= CALLING_OSI_PRESENTATION_SELECTOR= CALLING_OSI_AE_QUALIFIER= CALLING_OSI_AP_TITLE= CALLED_OSI_SESSION_SELECTOR= CALLED_OSI_PRESENTATION_SELECTOR= CALLED_OSI_AE_QUALIFIER= CALLED_OSI_AP_TITLE=	OSI communication parameters Note: This INI entry cannot be configured in the graphical user interface for the ICCP-TASE.2 Process gateway.

[VARIABLES]

Group for configured references between zenon variables and ICCP variables. The following are listed:

- ▶ Parameters of each ICCP-variable
- ▶ Number of variables
- ▶ Names of the variable

Entry	Description
PARAM_0=	<p>Parameter for ICCP-name.</p> <p>The numbering starts with 0. The counter is increased by 1 with each further parameter. The identifier corresponds to the identifier of the variable name.</p> <p>PARAM_0 corresponds to NAME_0.</p> <p>Example: PARAM_0=1, 0, 0, ICCP_Name</p>
COUNT=	Number of configured references
NAME_0=	<p>Name of the zenon variable.</p> <p>The numbering starts with 0. The counter is increased by 1 with each further parameter. This identifier corresponds to the parameter identifier.</p> <p>PARAM_0 corresponds to NAME_0.</p> <p>Example: NAME_0=MyName</p>

9.6 LOG entries

GENERAL

Parameter	Level	Note
Failed to get %s application object.	ERROR	Start zenon Runtime before you start Process Gateway.
Failed to start ICCP server.	ERROR	<p>Configure the settings of the Process Gateway.</p> <p>Configure a different port for each Process Gateway when starting several Process Gateways.</p>
<ul style="list-style-type: none"> ▶ Failed to interpret report for data set %s.%s. Bad message structure. ▶ Failed to interpret report for %s. Bad message structure. ▶ Failed to interpret write request for 	ERROR	Match the ICCP ' Data type ' and ' Extended type ' between Process Gateway and remote ICCP communication partners.

<pre>%s.%s. Bad message structure.</pre> <p>► Failed to interpret received data for %s.</p>		
<pre>Failed to write '%s', error code: %u.</pre>	ERROR	<p>The remote ICCP server responds to the write request in the negative, with MMS error code %u.</p> <p>Correct the settings on the remote server and/or move the variable from the client configuration to the server variables in Process Gateway.</p>

AUTOMATIC ESTABLISHMENT OF A CONNECTION

Parameter	Level	Description
<pre>Could not resolve host name %s. Errorcode: %d</pre>	ERROR	<p>The pre-set host name or the IP address could not be resolved.</p> <p>► %1 Host name or IP address of the set counterparty.</p> <p>► %2 Windows socket error code</p>
<pre>Autoconnection is not possible because no hostname (or IP address) is defined</pre>	ERROR	<p>The pre-set host name or the IP address are resolved to the address 0.0.0.0 and are thus invalid.</p>
<pre>Maximum number of %1 connections reached. Declining new connection from %2.</pre>	MSG	<p>Rejection of an incoming connection that has already achieved the maximum number of possible connections:</p> <p>► %1 Maximum number of connections could be established.</p> <p>► %2 IPv4 address of the counterparty.</p>
<pre>Cannot verify selected host name %1. Declining new connection from %2.</pre>	MSG	<p>The configured host name cannot be resolved to the IP address.</p> <p>► %1 Configured host name</p> <p>► %2 IPv4 address of the counterparty.</p>
<pre>Only connection from %1 allowed. Declining new connection from %2.</pre>	MSG	<p>An incoming connection is rejected because the source address does not correspond to the IP address of the configured host for automatic establishment of a connection.</p>

		<ul style="list-style-type: none"> ▶ %1 configured IP address of the set host. ▶ %2 IPv4 address of the counterparty.
Only one active connection allowed. Declining new connection from %2.	MSG	<p>An incoming connection is rejected because a connection has been established automatically.</p> <ul style="list-style-type: none"> ▶ %2 IPv4 address of the counterparty.
Creating a socket connection to the ICCP server %s [%2:%3].	DEBUG	<p>A TCP socket connection to the configured counterparty is established.</p> <ul style="list-style-type: none"> ▶ %1 Host name or IP address of the set counterparty. ▶ %2 IPv4 address of the counterparty. ▶ %3 Port number of the counterparty.
Could not create the socket connection to the ICCP server %1. Errorcode: %2.	DEBUG	<p>A TCP socket connection to the set counterparty could not be established within the socket timeout.</p> <ul style="list-style-type: none"> ▶ %1 Host name or IP address of the set counterparty. ▶ %2 Windows socket error code.

STATUS VARIABLES OF THE CONNECTION

Parameter	Level	Description
Server state variable with name '%s' does not exist or has an invalid type.	ERROR	<p>The configured status variable to display the status of the server connection does not exist in the project or has an invalid data type (required data type: numeric).</p> <ul style="list-style-type: none"> ▶ %s Configured name of the status variable.
Client state variable with name '%s' does not exist or has an invalid type.	ERROR	<p>The configured status variable to display the status of the client connection does not exist in the project or has an invalid data type (required data type: numeric).</p> <ul style="list-style-type: none"> ▶ %s Configured name of the status variable.
set server state: %1(%2)	MSG	<p>New status of a pre-existing server connection.</p> <ul style="list-style-type: none"> ▶ %1 Numeric value of the current status of the server connection

		<ul style="list-style-type: none"> ▶ %2 Current status of the server connection as a text (IDLE, CONNECTING, CONNECTED, ERROR)
set client state: %1(%2)	MSG	<p>New status of a pre-existing client connection.</p> <ul style="list-style-type: none"> ▶ %1 Numeric value of the current status of the client connection ▶ %2 Current status of the client connection as a text (IDLE, CONNECTING, CONNECTED, ERROR)

DISPLAY OF VARIABLES NOT PRESENT

Parameter	Level	Description
Unavailable server items variable with name '%1' does not exist or has an invalid type.	ERROR	<p>The configured status variable to display the name of the variable that does not exist does not exist in the project or has an invalid data type (required data type: STRING).</p> <ul style="list-style-type: none"> ▶ %1 Configured name of the status variable.
Error configured client variable '%1' is not available on the remote device.	ERROR	<p>A variable configured in Process Gateway for the client is not available on the remote ICCP server.</p> <ul style="list-style-type: none"> ▶ %1 ICCP name of the variable that is not present on the remote server

Parameter	Level	Description
Response variable [iccp name] already configured.	ERROR	<p>Error message for error projecting in the Process Gateway.</p> <p>With this error message, an ICCP client was assigned more than one SCADA variable. The first Response Object is applied, all other Response Objects are discarded.</p>
No ICCP object found for configured response variable [iccp name]	ERROR	<p>Error message for error projecting in the Process Gateway.</p> <p>With this error message an ICCP client was assigned a SCADA variable for a Response Object however the assignment of a matching Device object is missing.</p>

9.7 Interoperability

Process Gateway for ICCP-TASE.2 based on the **IEC 60870-6 TASE.2 ICCP** protocol.

Supported protocol versions: 2000.8 and 1996.8.

ICCP-TASE.2 supports Conformance Blocks 1 and 2 as client and server. As client Direct Control of Block 5 is supported.

The Process Gateway acts like a server by default when establishing communication.

10. IEC870 Slave

LANGUAGE

The IEC870SI Process Gateway and its dialogs for configuration are only available in English.

STRUCTURE

You can configure several virtual controllers (**devices**) in the IEC870 slave gateway. According to the used protocol, these virtual devices are associated directly to a Master (870-104) or polled by a Master (870-101). The **devices** can contain several sectors. These sectors contain IOs (Information Objects) representing the actual variables. An IO refers to a variable of the process control system.

OPERATION OF SEVERAL IEC870 SLAVES

IEC870SI Process Gateway can be started on both a standalone computer as well as in the zenon **network** (server/standby/client). All Process Gateways that are started in the network provide the same process image, but to the extent that is given in the configuration file.

Network servers are usually computers with a higher load. In large and complex systems, it may be advisable to operate the gateways on (dedicated) network clients. This has another benefit: with redundancy switching, a network client automatically connects to the respective primary server and continues to provide the data to higher-level systems.

Several instances of the IEC870 slaves can use the same or very similar configurations that have been transferred to several files, such as with export/import (on page 191) for example.

It is also possible to start several instances of the IEC870 slave on one computer.
requirement: separate communication partner for each instance - no conflicts with IP or COM ports.

REDUNDANCY OF THE IEC870 SLAVE

If two Process Gateways with the same configuration have been started then they are like a redundant slave for an 870 master, i.e. a redundant connection to the zenon system.

Two Process Gateways with the same configuration can be started on two computers in the zenon **network**, or on one computer with 2 Ethernet cards, or with different IP ports.

In the messaging direction, both IEC870 slaves provide the same Events and measured values, because zenon Runtime provides exactly the same process image. In command direction, one of the two slaves can be set to `silent mode` via an IO with the internal type T00 and IOA 2 (on page 210). The commands of the 870 master are thus only forwarded via a connection to Runtime.

The redundancy at Data Layer - according to the procedures of Edition 2 of the IEC 60870 standard, is not supported.

IDENTIFICATION AND ADDRESSING

For each respective started instance of the IEC870SI Process Gateway:

- ▶ 870-101: The virtual controller (**device**) is identified by the link address of the Slave.
- ▶ 870-104: The virtual controller is identified by the IP address of the master.
- ▶ Sector: Is addressed via COA (Common Object Address/Common Address of ASDU).
- ▶ IO: Is addressed by **Type Identification** and IOA.

In the following chapters, you will read how to configure the IEC870SI Gateway and how communication takes place.



Information

*For variables that are configured in the IEC870SI Process Gateway as command variables, each command is logged as **Send value** via the API interface during Runtime in the CEL if the **Write set value via VBA** property is active (**Logging in CEL** group).*

10.1 Module configuration

You have many settings options available in the IEC870 slave process gateway. The configuration of the IEC870SI is stored in a XML file (on page 191).

The lists in the configuration dialogs can generally be sorted. To change the sorting sequence, simply click on the column title.

870-101 PROTOCOL VS 104 PROTOCOL

The gateway supports both protocol IEC 60870-5-101 (abbreviated to: 870-101, serial communication) as well as IEC 60870-5-104 (abbreviated to: 870-104, TCP/IP communication). The settings for protocols -101 and -104 differ.

In the main window **Settings**, you can choose the protocol you want to use and define general settings:

- ▶ (Protocol) 870-101 (on page 156)
Unlocks the settings for protocol 101 and locks other, irrelevant input fields.
- ▶ (Protocol) 870-104
Unlocks the settings for protocol 104 and locks other, irrelevant input fields.

DEVICES

Virtual devices (connections) are created and deleted in the **Devices** list in the main window. A **device** defines how the Process Gateway, as an IEC870 slave, should work for a certain 870 master. These **devices** are configured in the tabs. Several devices can be set up - for several masters.

The following tabs are available in the process gateway:

- ▶ **Device** (on page 161) - general settings and master IP address (-104) or link address of the slave (-101).
- ▶ **Sector** (on page 166) - settings valid for a Common Address of ASDU (COA). A device should contain at least one sector; and one sector should contain at least one Information Object (IO):
 - **Information Objects** - the list of the Information Object Address (IOA), their Type Identifications (TID) and the assignment to zenon variables.
- ▶ **Sequences of Events (SOE (on page 201))**
SOE according to procedures defined in IEC 60870-5-101: 7.4.11.3 "Transmission of sequences of events" (in a section of a data file).
- ▶ **104 settings** (on page 188)
Data Layer Timeouts and APDU counters



Information

Only the general properties are documented in this section. You can find the protocol-dependent configurations in the section for the respective protocol (101) or in the description of the respective device settings (101 and 104).

TABS FOR THE CONFIGURATION OF THE DEVICES

Settings

Settings

Configuration file
IEC870SLAVE.XML Import... Export...

☒ 870-104 ☐ 870-101 unbalanced ☐ 870-101 balanced

IP Port
2404

Link Address Size (870-101)
1

COM Port
COM 1

Baud Rate
9600

☐ Deactivate standard DPI/DCS mapping and use internal mapping (not recommended).

Device	Name
0.0.0.1	Device 1

Add Delete

Device **Sector** **SOE** **104 settings**

IP-Address of Master (807-104) 0 . 0 . 0 . 1 Link Address of Slave (870-101) 1

Name / Description
Device 1

Short Pulse 1000 ms Long Pulse 2000 ms

Max. APDU size
253

☒ Commands allowed

Directory for File Transfer

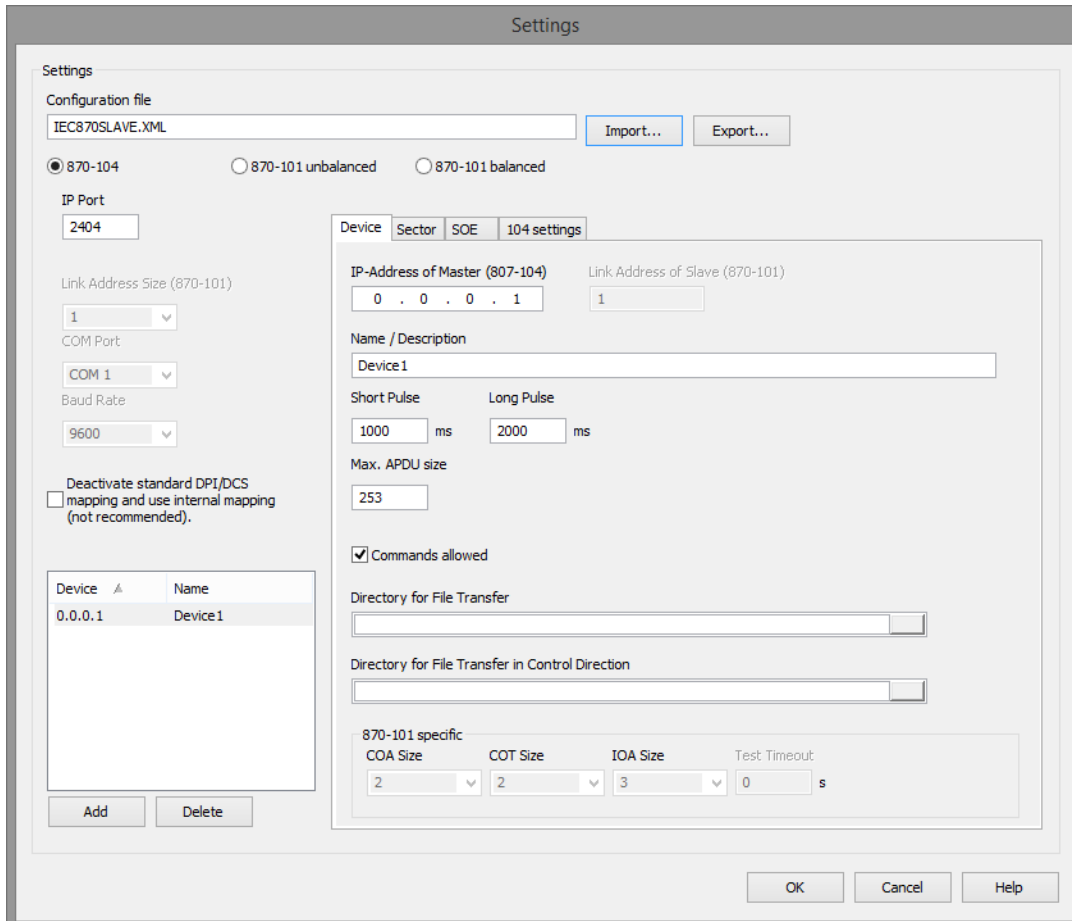
Directory for File Transfer in Control Direction

870-101 specific
COA Size 2 COT Size 2 IOA Size 3 Test Timeout 0 s

OK Cancel Help

Parameter	Description
Device	<p>Tab for the configuration of a device (on page 161).</p> <p>A virtual device corresponds to an interface with an 870 master.</p>
Sector	<p>Tab for the configuration of the sectors (on page 166) of the device.</p> <p>A sector corresponds to a Common Object Address/Common Address of ASDU and is addressed via COA .</p> <p>Note: A sector can contain several Information Objects that are linked to zenon variables and addressed with Type Identification (Type ID) and Information Object Address (IOA).</p>
SOE	<p>Tab for the configuration of the recording and transfer of the SOE files (on page 201) (Sequences of Events)</p>
104 settings	<p>Tab for the configuration of the Protocol-870-104-specific settings for timeouts (on page 188) and APDU counter.</p>
OK	<p>Accepts all changed settings. They will immediately be stored in the current configuration file (as defined in the Configuration file).</p> <p>Attention: If you have not made any changes, close the dialog with the Cancel, not OK.</p> <p>After clicking on OK, all existing connections with the 870 master will be closed and restarted with the new settings.</p>
Cancel	<p>Discards all changes made and closes the dialog window.</p> <p>The dialog is closed without the pre-existing communication to 870 masters being interrupted.</p>
Help	<p>Opens online help.</p>

10.1.1 General Settings



Settings

Configuration file
IEC870SLAVE.XML Import... Export...

☒ 870-104 ☐ 870-101 unbalanced ☐ 870-101 balanced

IP Port
2404

Link Address Size (870-101)
1

COM Port
COM 1

Baud Rate
9600

☐ Deactivate standard DPI/DCS mapping and use internal mapping (not recommended).

Device	Name
0.0.0.1	Device 1

Add Delete

104 settings

IP-Address of Master (807-104)
0 . 0 . 0 . 1

Link Address of Slave (870-101)
1

Name / Description
Device 1

Short Pulse
1000 ms

Long Pulse
2000 ms

Max. APDU size
253

☒ Commands allowed

Directory for File Transfer

Directory for File Transfer in Control Direction

870-101 specific

COA Size
2

COT Size
2

IOA Size
3

Test Timeout
0 s

OK Cancel Help



Information

Only the general properties of the IEC870 Slave are documented in this section. You can find further information in the description of the virtual **device**.

SETTINGS

General settings:

Parameter	Description
Configuration file	<p>Name of the XML configuration file with the configuration of the IEC870 slave in the Process Gateway.</p> <p>This is where all active settings are stored and also loaded from. The name you enter must be a valid file name. Otherwise, no data can be stored. In this case, you will get an error message.</p> <p>Only enter the file name without path. You do not have to add the file extension XML, as it will be added automatically during saving. The current INI directory will be used as path.</p> <p>Save location: %CD_SYSTEM%</p> <p>You can find the INI file in the system folder of the process control system. It is called '<SCADA>ProcGateway.ini' - <SCADA> is the placeholder for the name of the process control system.</p> <p>The file name will automatically be entered in the INI file of the process gateway as configuration.</p> <p>You can find the description of the structure of the XML file in the XML file (on page 191) chapter.</p>
Import...	<p>Loads a project configuration (on page 197) from an XML or CSV file.</p> <p>The current settings are replaced by the ones stored in the file. The current setting is retained for Configuration file.</p> <p>Clicking on the ... button opens the file selection dialog to select a configuration file.</p> <p>Note: You can thus transfer the configurations of other projects or import the configurations with large amounts to variables that you have prepared and exported with other applications.</p> <p>The file content is checked during import. A corresponding warning dialog is shown in the event of an error.</p>
Export...	<p>Exports current project configuration (on page 197) as an XML or CSV file.</p> <p>Clicking on the ... button opens the file selection dialog to select a save location for the target file.</p> <p>Attention: A Device that still has the IP or link address issued on creation is not exported. To export it, you must first configure the connection to the 870 master in the Device (on page 161) configuration window.</p> <p>Note:</p> <ul style="list-style-type: none"> ▶ No file is exported if there are no IOs in the current Process Gateway configuration. ▶ The configured IP address is exported for 870-104 connections. ▶ The configured Link address is exported for 870-101.

870-104	<p>activates protocol 104, unlocks all input fields for 104 and locks settings for 101.</p> <p>In this mode, several slaves can communicate with several masters via the TCP/IP interfaces. Each TCP socket defines a 1:1 connection between slave and master; it is thus possible to transfer the value changes to the slave spontaneously. The loss of the connection can be detected both by the slave and master thanks to the test procedures running in the background (replacement of small APDUs on the Link Layer).</p>
870-101 unbalanced	<p>Activates protocol 101 in <i>unbalanced Mode</i>, unlocks all specific settings for 101 and locks settings for 104.</p> <p>In this mode, several slaves can communicate with several masters via the serial interface. A slave should only transfer the data to a master on one (cyclical) Link Layer request, otherwise conflicts could occur in the Link Layer. There is no test procedure for slaves to check whether the connection to the master continues to exist.</p>
870-101 balanced	<p>Activates protocol 101 in <i>balanced Mode</i>, unlocks all specific settings for 101 and locks settings for 104.</p> <p>In this mode, only a slave can communicate with a master using the serial interface. It allows the slave to transfer the value changes spontaneously. The loss of the connection can be detected both by the slave and master thanks to the test procedures running in the background (replacement of small APDUs on the Link Layer).</p>
IP Port	<p>IP Port for communication via IEC 60870-5-104 protocol. Slave port to which the masters are to connect. The protocol defines this as fixed at 2404.</p> <p>Default: 2404</p> <p>Allowed port numbers: 1200 to 65535</p> <p>Note: Applicable for all connections for protocol 104. Inactive if 870-101 unbalanced or 870-101 balanced has been selected as a protocol.</p> <p>It is possible to have several instances of the Process Gateway running at the same time. That is why you can change this port.</p>
Link Address Size (870-101)	<p>Size of the link layer address, in bytes, for communication via the IEC 60870-5-101 protocol. Select from drop-down list.</p> <p>Default 1</p> <p>Note: The same size is applicable to all connections to 870-101 masters. Inactive if 870-104 is selected as a protocol.</p>
COM Port	<p>Serial interface of the computer with zenon Runtime. Select from drop-down list.</p> <p>Default 1</p> <p>Note: Applicable to all connections to 870-101 masters (<i>balanced</i> and <i>unbalanced</i>). Inactive if 870-104 is selected as a protocol.</p>

Baud Rate	<p>Data transfer rate. Select from drop-down list. Default 9600</p> <p>Note: Applicable to all connections to 870-101 masters (balanced and unbalanced). Inactive if 870-104 is selected as a protocol.</p>
Deactivate standard DPI/DCS mapping and use internal mapping	<p>Deactivates the standard communication via DPI/DCS and uses internal mapping in communication. The non-compliant behavior is disabled when this option is activated.</p> <p>Default: Inactive</p> <p>Attention: Activation of this option is not recommended.</p>

10.1.2 Device List

The virtual devices in the IEC870 slave - **Devices** - are created and deleted in the left area of the configuration dialog.

The details are configured in the right area of the dialog with the corresponding tabs.

To change the configuration of an existing **device**, select the desired **Device** in the list. The attendant configurations can then be amended in the right area.

Parameter	Description
Device	<p>Identification of the device: IP address of the master (104) or link address of the slave (101).</p> <p>Default: ? ? ? ? (Not defined)</p> <p>Attention: Before you can use or export the device, you must configure it.</p>
Name	<p>Name of the device/connection - desired text for easier identification.</p>
Add	<p>Adds a new device with standard settings. By default, it gets the invalid ID " ? ? ? ? " and invalid IP/link addresses.</p> <p>Attention: Before you can use or export the device, you must configure it.</p>
Delete	<p>Deletes the selected device. This button is not active if no device has been selected.</p> <p>Attention: The selected entry is deleted immediately without a request for confirmation beforehand.</p>

A device is created or deleted on the left-hand side of the dialog.

Clicking on the **Add** button adds a device, which is configured in the **Device**, **Sector**, **SOE** and **104 settings** tabs. Several devices can be created. A simple click on the respective device in the list opens the attendant configuration.

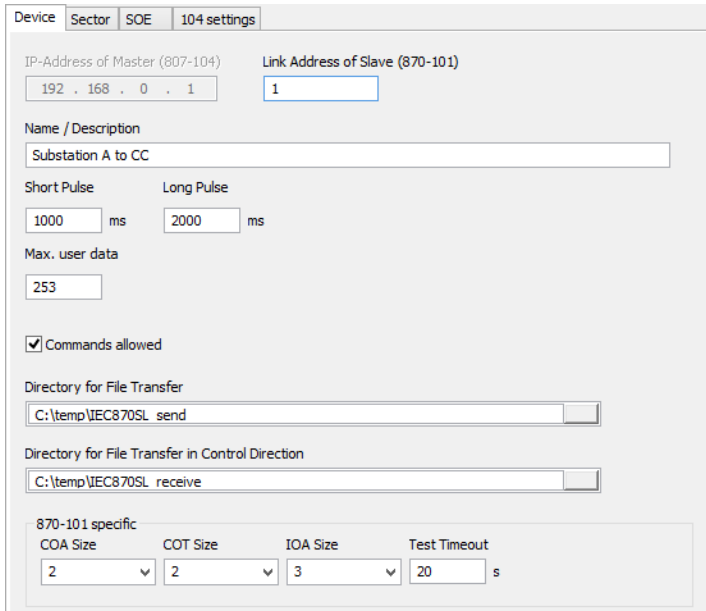
VALIDATION

Configurations are validated by clicking on the **OK** button. A corresponding warning dialog is shown in the event of an error. The check takes the selected connection into account (870-104 or 870-101). If the configuration is transferred from a file, the file content is also taken into account during validation.

When the IEC870 Process Gateway starts, there is also an initial check of the existing configuration.

10.1.3 Device - Configuration

You define the settings of the virtual device in the **Device** tab in the main window. Depending on the protocol selected - 870-104 or 101 - certain areas are grayed out. A precise definition of which input fields are available for which protocol is in the list below.



Device			
Sector	SOE	104 settings	
IP-Address of Master (807-104)		Link Address of Slave (870-101)	
192 . 168 . 0 . 1		1	
Name / Description			
Substation A to CC			
Short Pulse		Long Pulse	
1000 ms		2000 ms	
Max. user data			
253			
<input checked="" type="checkbox"/> Commands allowed			
Directory for File Transfer			
C:\temp\IEC870SL_send			
Directory for File Transfer in Control Direction			
C:\temp\IEC870SL_receive			
870-101 specific			
COA Size	COT Size	IOA Size	Test Timeout
2	2	3	20 s

Parameter	Protocol	Description
IP-Address of Master (807-104)	104	<p>IP address of the master, which is permitted to establish a connection to the slave. The device that the master communicates with is detected using the IP address. All IP addresses except 255.255.255.255 are allowed.</p> <p>The configured IP address is shown in the list of Devices in the Device column.</p> <p>Default: 255.255.255.255</p> <p>Note: The current value is lost if the protocol is changed.</p>
Link Address of Slave (870-101)	101	<p>The Link Layer address of the slave. The master uses the link address contained in the TF 1.2 frame to specify the device for which the telegram is applicable. With link addresses, a Master can communicate with multiple devices. The allowed range of the address depends on the Link Address Size setting in the main settings.</p> <ul style="list-style-type: none"> ▶ Link address size = 0 Only applicable for balanced Mode. No range permitted. The connection is unique (1 Slave : 1 Master) ▶ Link address size = 1 Permitted range: 0 bis 254 ▶ Link address size = 2 Permitted range: 0 bis 65534 <p>Default: -1 (invalid)</p> <p>Note: Configuration of this option is not necessary if 870-101 balanced has been selected.</p> <p>The current value is lost if the protocol is changed.</p>
Name/Description	101 104	<p>Name of the device for easy identification, a desired text.</p> <p>The configured name is displayed in the list of Devices in the Name column.</p>
Short pulse	101 104	<p>Defines the length of the pulse for the Qualifier of Command (on page 219) in the 'short pulse duration' (QOC = 1) version in milliseconds.</p> <p>Default: 1000 ms</p> <p>Valid values: 0 bis 4294967295</p>
Long pulse	101 104	<p>Defines the length of the pulse for the Qualifier of Command (on page 219) in the 'long pulse duration' (QOC = 2) version in milliseconds.</p> <p>Default: 2000 ms</p> <p>Valid values: 0 to 4294967295.</p>

Max APDU size	104	Maximum length of APDU data as defined in the standard IEC 60870-5-104. Default: 253 Valid values: between 25 and 253.
Max. user data	101	Maximum length of the User Data as defined for TF1.2 in IEC60870-5-2 Section 3.2. Default: 253 Valid values: between (19 + LAF size + COA size + COT size + IOA size) and (255).
Commands allowed	101 104	This option allows commands from the master to be prevented on a lasting basis. The commands are not accepted if the setting is deactivated. Corresponding entries are created in the LOG file. Default: active Note: The commands can be dynamically activated or deactivated using a variable in zenon (T00, IOA 2) (on page 210).
Directory for File Transfer	101 104	Defines the directory for transfer of files and directories. Clicking on the ... button opens the file selection dialog to select a source directory. Note: The files are sent to the master.
Directory for File Transfer in Control Direction	101 104	Defines the directory in which the files for transfer of files and directories are stored. Clicking on the ... button opens the file selection dialog to select a target directory. Note: These files are received by the master. All files with the NOF 255 (the temporary NOF for reverse file transfer) are ignored for the directory query.

870-101 SPECIFIC

Parameter	Protocol	Description
COA size	101	Defines the length of the COA (Common Object Address/Common Address of ASDU). Selection of address size from drop-down list: <ul style="list-style-type: none"> ▶ 1 octet ▶ 2 octets ▶ Note: If the 60870-5-104 (TCP/IP) connection type is selected for link layer, the value 2 octets is expected in accordance with the standard.



		Default: 2
--	--	------------

COT size	101	<p>Defines the length of the COT (Cause of Transmission).</p> <p>Selection of address size from drop-down list. Valid:</p> <ul style="list-style-type: none"> ▶ 1 octet ▶ 2 octets ▶ Note: If the 60870-5-104 (TCP/IP) connection type is selected for link layer, the value 2 octets is expected in accordance with the standard. <p>Default: 2</p>
IOA size	101	<p>Defines the length of the IOA (Information Object Address).</p> <p>Selection of address size from drop-down list. Valid:</p> <ul style="list-style-type: none"> ▶ 1 octet ▶ 2 octets ▶ 3 octets ▶ Note: If the 60870-5-104 (TCP/IP) connection type is selected for link layer, the value 3 octets is expected in accordance with the standard. <p>Default: 3</p>
Test timeout	101	<p>Timeout time for 870-101 connections. Entry in seconds.</p> <ul style="list-style-type: none"> ▶ In 101 unbalanced: States the time (in seconds) in which a master must execute a query to the slave. If, within this time, there is no query from the master, the slave ends the connection. ▶ In 101 balanced: States the time (in seconds) in which the slave will execute the test procedures on the link layer. The test procedure is mandatory as prescribed by the IEC60870-5-101 standard for balanced. The slave can thus detect a loss of connection whilst there is no spontaneous transfer of data. <p>Default: 20</p> <p>Note: in unbalanced Mode , monitoring is deactivated in the event of a configured value of 0.</p> <p>A value greater than 0 must be configured in balanced Mode. In the event of inactivity (no data to send), a test frame is sent after the time configured here. This must be confirmed by the master.</p> <p>If an answer is missing, the last frame (test frame or spontaneous data) is repeated. If no response to it is received, the Process Gateway restarts communication with a Request Status of</p>

		Link.
--	--	-------

10.1.4 Sector - Configuration

The **Sector** tab lists all sectors configured in the **device**. If you select a sector from the drop-down list, its **Information Objects** (on page 172) (IOs) are displayed.

DeviceSectorSOE104 settings

Common Address of ASDU (COA)

<1> Sector1

NewEditDelete

Information Objects

IOA	Type ID	SCADA variable
1	<0> Internal status	iecT00_1_conn_st
1	<103> Clock synchronization	iecT103_enable
2	<0> Internal status	iecT00_2_comm_ena
3	<0> Internal status	iecT00_3_SOE
4	<0> Internal status	iecT00_4_SOE
5	<0> Internal status	iecT00_5_FT
6	<0> Internal status	iecT00_6_SOE
100	<1> Single point information	iecT01_00
100	<58> Single command with time tag CP56Time2a	iecT45_00
101	<2> Single point information with time tag	iecT01_01
101	<45> Single command	iecT45_01
102	<30> Single point information with time tag CP56Time2a	iecT01_02
102	<45> Single command	iecT45_02
103	<1> Single point information	iecT01_03
104	<2> Single point information with time tag	iecT01_04
105	<30> Single point information with time tag CP56Time2a	iecT01_05

NewEditDelete

COMMON ADDRESS OF ASDU (COA)

You configure the sectors per **device** in the upper area. The **COA** and the name of the sector are displayed.

Configured sectors are displayed in a drop-down list. The first sector of the drop-down list is always shown by default. Configured **Information Objects** are shown for this sector in the area below.

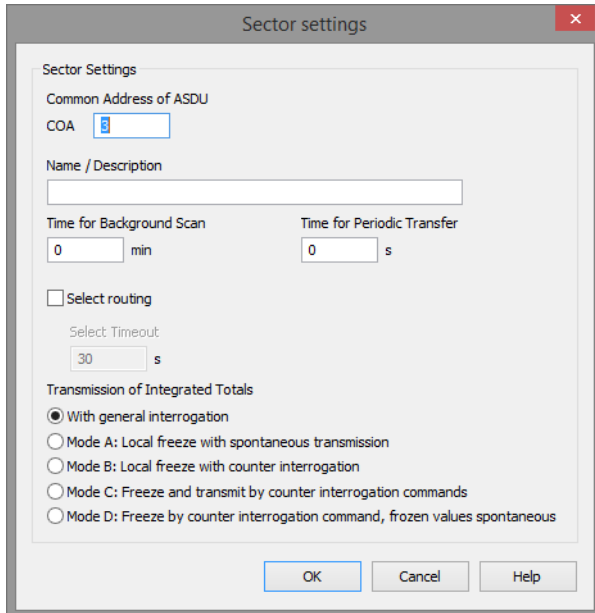
Parameter	Description
New	Opens the dialog to create a new sector (on page 168).
Edit	<p>Activates the dialog for editing the selected sector.</p> <p>Sectors that have already been configured are selected from the drop-down list.</p> <p>You can also launch the editor by double-clicking on the sector in the list.</p> <p>This button is inactive if no sector was selected.</p>
Delete	<p>Deletes the selected sector.</p> <p>This button is inactive if no sector was selected.</p> <p>Attention: the selected sector is deleted without a request for confirmation.</p>

INFORMATION OBJECTS

Information Objects (IOs) for sectors are administered in the lower area. The IOs with COA of the selected sector are shown.

Parameter	Description
List of the configured information objects per COA	<p>List of the configured IOAs for the selected COA.</p> <ul style="list-style-type: none"> ▶ IOA Configured address of the Information Objects. ▶ Type ID Configured Type Identification of the Information Object. ▶ SCADA variable Linked SCADA variable of the Information Object. <p>The list can be sorted by clicking on the column header. The column widths can be amended individually.</p>
New	<p>Activates the dialog for creating new IOs (on page 172).</p> <p>This subsequent dialog is closed and a new IO is created by clicking on 'OK'.</p> <p>This button is inactive if no sector was selected.</p>
Edit	<p>Activates the dialog for editing the selected IO.</p> <p>You can also launch the editor by double-clicking on the IO in the list.</p> <p>This button is inactive if no IO was selected.</p>
Delete	<p>Deletes the selected information object.</p> <p>This button is inactive if no IO was selected.</p>

Sector Settings



The image shows a 'Sector settings' dialog box with a title bar containing a close button. The dialog contains the following fields and options:

- Sector Settings** (Section Header)
- Common Address of ASDU**
 - COA:
- Name / Description**
 -
- Time for Background Scan**
 - min
- Time for Periodic Transfer**
 - s
- ☐ **Select routing**
 - Select Timeout:
 - s
- Transmission of Integrated Totals**
 - ☒ With general interrogation
 - ☐ Mode A: Local freeze with spontaneous transmission
 - ☐ Mode B: Local freeze with counter interrogation
 - ☐ Mode C: Freeze and transmit by counter interrogation commands
 - ☐ Mode D: Freeze by counter interrogation command, frozen values spontaneous

At the bottom of the dialog are three buttons: **OK**, **Cancel**, and **Help**.

Parameter	Description
COA	The Common Address of ASDU/Common Object Address (IEC 60870-5-101 7.2.4) by which the sector is addressed. This number must be unique for each device (1..254).
Name/Description	Any name. For easy identification of a sector.
Time for Background scan [min]	<p>The "Background scan" cycle in accordance with the requirements of the IEC 60870-5-101 standard, chapter 7.4.13 "Background scan". The values are sent with COT = 2 (background scan/COT_back). The time is stated in minutes.</p> <p>In the set time grid, there is a cyclical exchange of values for all IOs with the Background scan setting activated.</p> <p>Values are only transferred cyclically if no spontaneous value is transferred within the time set (no COT = 3).</p> <p>If the value is "0", cyclical transfer of values is deactivated for this sector.</p> <p>Default: 0</p>
Time for periodic transfer [s]	<p>Support of cyclical data transfer in accordance with IEC 60870-5-5 6.3. The values are sent with COT = 1 (periodic, cyclic / COT_per). The time is configured in seconds.</p> <p>The cyclical transfer of values is carried out for all IOs with the Periodic data transfer setting activated. These IOs are only transferred cyclically and not spontaneously.</p> <p>If the value is "0", cyclical transfer of values is deactivated for this sector.</p> <p>Default: 0</p> <p>Note: the quickly-fluctuating measured values can unnecessarily impair spontaneous transfer. We recommend however - instead of periodic data transfer - configuring Hysteresis for the respective IO.</p>
Select routing	<p>If active, a prior Select is necessary for an Execute when the command is executed. The switching commands from the master are forwarded using the command processing - via the auto/remote actions - to the respective Energy driver in zenon Runtime.</p> <p>Note: You can find further information in relation to this in the Select routing (on page 217) chapter.</p>
Select Timeout [s]	<p>Timeout for Select before Execute in seconds.</p> <p>The Select process must be ended within the configured time and an Execute must be triggered. If there is no confirmation of the Select from zenon Runtime or the Select has been confirmed but the master does not send an Execute, the Process Gateway cancels the process.</p> <p>Note: Inactive if Select routing is not active.</p> <p>Default: 30 s</p>

	Input range: 1 bis 65535 Note: You can find further information in relation to this in the Select routing (on page 217) chapter.
--	---

TRANSMISSION OF INTEGRATED TOTALS

Selection of the transfer mode (see IEC 60870-5-101 7.4.8) for the integrated totals - the IOs of type ID:

- ▶ 15 (M_IT_NA_1)
- ▶ 16 (M_IT_TA_1)
- ▶ 37 (M_IT_TB_1)

No Reset is supported; an FRZ with Reset creates a warning in the LOG.

Selection from options list.

Default: With general interrogation

For Mode A to Mode D, IOs are requested immediately after connecting to the master of zenon. Value changes to the Integrated Totals IOs are administered in a separate buffer. This can buffer the greater value from (number of the ITs IO) * 4, or 2048 value changes.

Parameter	Description
	Integrated Totals are also transferred during a general interrogation (C_IC_NA_1).

Mode A: Local freeze with spontaneous transmission	Integrated totals are transferred when a value is changed.
Mode B: Local freeze with counter interrogation	Integrated totals are only transferred with a Counter Interrogation Command (C_CI_NA_1). To do this, a maximum of five seconds is waited after the interrogation is activated until all ITs have been supplied with values. ITs without value - the zenon variable without a value - are transferred with the qualifier <i>invalid</i> . The value changes are buffered until the interrogation is completed (freeze active). All ITs thus transfer the value of the point in time, when the last IT was supplied with a value.
Mode C: freeze and transmit by counter interrogation Transmission of integrated totals commands	Same behavior as for Mode B .
Mode D: freeze by counter interrogation command, frozen values spontaneous	Transfer as with mode C, mode A is activated after conclusion of the interrogation.
OK	A check is carried out to see if the COA is free for the device by pressing the OK button. If it is already in use you will receive an error message. If it is available, the setting is accepted and the dialog is closed.
Cancel	Discards all entries and closes the dialog.
Help	Opens online help.

Information Objects

Information object settings

Settings

Information Object Address
IOA

Type Identification
<0> Internal status

SCADA Variable
 ...

Data Transfer
☒ Spontaneous
☐ Background Scan
☐ Periodic Data Transfer
☐ Off
☐ Spontaneous, buffered when master is offline

Sequence of Events (SOE)
☒ Inactive
☐ All Events
☐ Raising Edge
☐ Falling Edge
☐ Raising & Falling Edge
☐ File Trigger

Hysteresis
Negative Positive

OK Cancel Help

SETTINGS

Parameter	Description
Information Object Address	<p>Defines the address (IOA) under which the IO is addressed (IEC 60870-5-101 7.2.5).</p> <p>Default: 1</p> <p>The IOA must be unique in each monitoring and controlling direction per sector.</p> <p>Example:</p> <ul style="list-style-type: none"> ▶ Valid addressing: Two IOs, both with the same IOA, a T01 (single-point - M_SP_NA_1) and a T45 (singlecommand - C_SC_NA_1). The addressing is recommended if this single command checks precisely the single-point information. ▶ Invalid addressing: Two IOs, both with the same IOA, a T01 and a T15 (integrated totals - M_IT_NA_1). This addressing is not permitted because both type IDs belong to the monitoring direction.
Type identification	<p>Drop-down list with the supported Type identifications (IEC 60870-5-101 7.2.1).</p> <p>Default: <0> Internal status</p> <p>This setting is also called type ID.</p> <p>It also determines the variables of the process control system that can be associated to the IO based on their data type.</p> <p>Note: You can also find more detailed information on type ID in the chapters Assignment of data types (on page 179) and Interoperability (on page 232).</p>
SCADA variable	<p>Defines the variable from the process control system associated with the IO. Syntax: <Project name>#<variable name> define the project connection. If there is no project reference for a variable, it is assumed that this variable is present in the start project.</p> <p>While entering, the system checks if the specified variable exists in the corresponding project and if it has a data type compatible with the type identification.</p> <p>The Button '...' opens the dialog for variable selection.</p> <p>Read more about this in the section 'Variable selection (on page 178)'. (Scada stands for the name of the process control system.)</p> <p>Supported data types:</p> <p>BOOL; BYTE; WORD; DWORD; LWORD; SINT; INT; DINT; LINT; USINT; UINT; UDINT; ULINT; REAL; LREAL; long double; Real number IEEE</p>

DATA TRANSFER

Determines the type of data transfer for the selected IO Selection from option list.

Default: Spontaneous

Parameter	Description
Spontaneous	Data transfer of the IOs is spontaneous - if the value of the linked SCADA variable changes.
Background Scan	<p>IOs are transferred as a Background Scan in accordance with to the IEC 60870 standard. A data point that does not receive a new spontaneous value within the set Time for Background Scan is automatically sent by the gateway after the time has expired, with COT = 2.</p> <p>Values are only transferred if no spontaneous value (COT = 3) for this IO has yet been transferred within the time set.</p> <p>The setting is inactive if the background scan is locked in the sector (Time=0) or if the type ID used is not suitable for a background scan in accordance with the protocol. Permitted type IDs:</p> <ul style="list-style-type: none"> ▶ M_SP_NA_1 and M_SP_TA_1, M_SP_TB_1 - for T01, ▶ M_DP_NA_1 and M_DP_TA_1, M_DP_TB_1 - for T03, ▶ M_ST_NA_1 and M_ST_TA_1, M_ST_TB_1 - for T05, ▶ M_BO_NA_1 and M_BO_TA_1, M_BO_TB_1 - for T07, ▶ M_ME_NA_1 and M_ME_TA_1, M_ME_TD_1 - for T09, ▶ M_ME_NB_1 and M_ME_TB_1, M_ME_TE_1 - for T11, ▶ M_ME_NC_1 and M_ME_TC_1, M_ME_TF_1 - for T13 <p>Note: The type IDs with time stamp (for example M_ME_TA_1) is mapped to the corresponding TID without time stamp (for example M_ME_NA_1) in the process of sending the background scan.</p> <p>The setting is not available for any other type ID.</p>
Periodic Data Transfer	<p>Periodic data transfer in accordance with the IEC 60870-5-5 Standard, Section 6.3.</p> <p>IOs are transferred cyclically and not spontaneously with COT = 1.</p> <p>The setting is inactive if periodic data transfer is locked for the sector (Time for periodic transfer=0) or if the type ID used does not support periodic data transfer.</p> <p>Only the type ID=9, 11 and 13 support a periodic transfer - M_ME_NA_1, M_ME_NB_1, M_ME_NC_1 - "measured values" without time stamp.</p> <p>The setting is not available for any other type ID.</p> <p>Note: the quickly-fluctuating measured values can unnecessarily impair spontaneous transfer. We recommend however - instead of periodic data transfer - configuring Hysteresis for the respective IO.</p>
Off	Changes in value are not transferred. No value is sent for the IO during a

	general interrogation (GI).
Spontaneous, buffered when master is offline	<p>When a connection fails, the data is buffered until the master connects again. All buffered data is sent with the status COT_spont (3). The last buffered value is sent with the status COT_inrogen (20) again.</p> <p>Buffering can be activated for the type IDs <30..37>:</p> <p>M_SP_TB_1, M_DP_TB_1, M_ST_TB_1, M_BO_TB_1, M_ME_TD_1, M_ME_TE_1, M_ME_TF_1 und M_IT_TB_1</p> <p>Note: Only type IDs that have a time stamp with a date are permitted as Events ("time tag CP56Time2a").</p> <p>The setting is not available for any other type ID.</p>

SEQUENCES OF EVENTS (SOE)

These settings define the IO-specific behavior for SOE. The saving of the SOE files and the transfer thereof (see IEC standard 60870-5-101 7.4.11.3.1) can only be activated for the type IDs <30..37>:

M_SP_TB_1, M_DP_TB_1, M_ST_TB_1, M_BO_TB_1, M_ME_TD_1, M_ME_TE_1, M_ME_TF_1 und M_IT_TB_1.

Default: Inactive

Note: Only type IDs that have a time stamp with a date are permitted as Events ("time tag CP56Time2a").

Parameter	Description
Inactive	Data point is not used for SOE description. Is automatically used for all type IDs that do not support SOE.
All events	Each value or status change triggers a write in the open file. The setting is blocked if the type ID is not planned for the SOE function.
Raising Edge	If the value changes from 0 to $\neq 0$, then it triggers the write mode in the open file. The setting is blocked if the type ID is not planned for the SOE function.
Falling Edge	If the value changes from $\neq 0$ to 0, then it triggers the write mode in the open file. The setting is blocked if the type ID is not planned for the SOE function.
Raising + Falling Edge	A change of value from 0 to $\neq 0$ or from $\neq 0$ to 0 triggers a write in the open file. The setting is blocked if the type ID is not planned for the SOE function.
File trigger	If a value is to be written due to the settings previously described, a new file is created and opened for the activated file, provided this has not yet been opened. The trigger event itself is also written to the SOE file. The setting is blocked if the type ID is not planned for the SOE function. Default: Inactive

HYSTERESIS

Value changes that are within the hysteresis are ignored - not transferred to the master.

Hysteresis is applicable for the permitted **TID** ("measured values") and only if these have been configured for spontaneous data transfer.

A transfer is triggered:

- ▶ If **Type Identification** allows hysteresis.
The **Type Identifications** <9> to <14> and <34> to <36> are permitted.
- ▶ If **Data Transfer** = spontaneous
- ▶ With a value change that acts \geq the hysteresis (according to the direction).

And notwithstanding the hysteresis:

- ▶ Change of the quality (status bits of the variable) always triggers the transfer;
- ▶ The current value is always sent to a GI (**C_IC_NA_1**) and a read (**C_RD_NA_1**) command.

Parameter	Description
Negative	<p>Negative hysteresis, starting from measuring range of the variable.</p> <p>Default: 0</p> <p>Note: not active if type identification is not <9> to <14> or <34> to <36>. This field is also inactive if the value Data Transfer is not spontaneous.</p> <p>Erroneous and meaningless entries (such as text input or incorrect hysteresis values) are ignored and must be corrected by the engineer of the project.</p>
Positive	<p>Positive hysteresis, starting from measuring range of the variable.</p> <p>Default: 0</p> <p>Note: not active if type identification is not <9> to <14> or <34> to <36>. This field is also inactive if the value Data Transfer is not spontaneous.</p> <p>Erroneous and meaningless entries (such as text input or incorrect hysteresis values) are ignored and must be corrected by the engineer of the project.</p>
OK	Your settings are verified after pressing the button "OK". If they are valid, they are accepted and the dialog is closed.
Cancel	Discards all entries and closes the dialog.
Help	Opens online help.

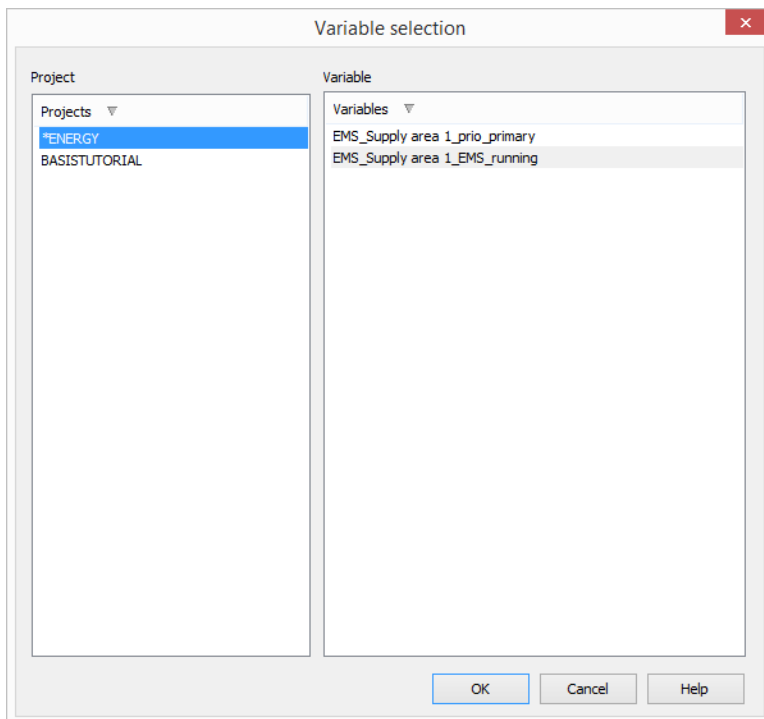


Information

*Communication for **Spontaneous data transfer**, **Background Scan** and **Periodic Data Transfer** only starts after the first general query received (`C_IC_NA_1`). For background scan and periodic data transfer, the start of the transfer cycle is calculated per IO from the time of the answer to the general query.*

Variable selection dialog

This dialog lists all available projects and their variables.



Parameter	Description
Projekte	List of the projects started in zenon Runtime. A star (*) marks the default project.
Variablen	The list of variables is filtered. Only variables that can be used with the selected Type identification for the IO are displayed. Selecting a project updates the list of variables.
OK	Assumes the selected variable. Alternatively, the variable can be assumed by double clicking. When loading, the currently configured project and the variable will be selected. The same variable can be used several times: for several IOs in the same sector or other sectors, for the same device or another one. If the dialog remains open and there is a reload in the Runtime, all lists will be emptied and refilled after the reload is finished.
Cancel	Discards all entries and closes the dialog.
Help	Opens online help.

Mapping of the data types

The zenon data types are compared to the IEC 60870 data types in the following table.

Process Gateway		zenon	
IEC 60870-5-101 ASDU	Type identification	Data type	Comment
M_SP_NA_1	1	BOOL	
M_SP_TA_1	2	BOOL	
M_SP_TB_1	30	BOOL	
M_DP_NA_1	3	USINT	0 . . 3, with DPI/DCS Mapping
M_DP_TA_1	4	USINT	0 . . 3, with DPI/DCS Mapping
M_DP_TB_1	31	USINT	0 . . 3, with DPI/DCS Mapping
M_ST_NA_1	5	USINT	Corresponds to complete VTI (IEC60870-5-101 7.2.6.5); the highest bit in USINT is thus the Transient bit.
M_ST_TA_1	6	USINT	Corresponds to complete VTI (IEC60870-5-101 7.2.6.5); the highest bit in USINT is thus the Transient bit.
M_ST_TB_1	32	USINT	Corresponds to complete VTI (IEC60870-5-101 7.2.6.5); the highest bit in USINT is thus the Transient bit.
M_BO_NA_1	7	UDINT	
M_BO_TA_1	8	UDINT	
M_BO_TB_1	33	UDINT	
M_ME_NA_1	9	REAL	<p>The value range in accordance with the IEC 60870 standard is $<-1 \dots 1-2^{-15}>$ with a precision of 4 decimal places. With zenon variables, this corresponds to a value range of -1.0 to 0.9999.</p> <p>Outside of the range for normalized values, in ASDU the value of the IOs is amended and the OV quality bit is set.</p>
M_ME_TA_1	10	REAL	<p>The value range in accordance with the IEC 60870 standard is $<-1 \dots 1-2^{-15}>$ with a precision of 4 decimal places. With zenon variables, this corresponds to a value range of -1.0 to 0.9999.</p> <p>Outside of the range for normalized values, in ASDU the value of the IOs is amended and the OV quality bit is set.</p>
M_ME_TD_1	34	REAL	<p>The value range in accordance with the IEC 60870 standard is $<-1 \dots 1-2^{-15}>$ with a precision of 4 decimal places. With zenon variables, this corresponds to a value range of -1.0 to 0.9999.</p> <p>Outside of the range for normalized values, in ASDU the value of the IOs is amended and the OV quality bit is set.</p>

M_ME_NB_1	11	INT	
M_ME_TB_1	12	INT	
M_ME_TE_1	35	INT	
M_ME_NC_1	13	REAL	
M_ME_TC_1	14	REAL	
M_ME_TF_1	36	REAL	
M_IT_NA_1	15	DINT	
M_IT_TA_1	16	DINT	
M_IT_TB_1	37	DINT	
M_EI_NA_1	70	USINT	A value change from 126 to 127 causes End of initialization (on page 208) to be sent to the master
C_SC_NA_1	45	BOOL	The variable reflects the value of the command that has been received from the 870 master and the status bits of the variable reflect the process of the command. Example: COTx.
C_SC_TA_1	58	BOOL	The variable reflects the value of the command that has been received from the 870 master and the status bits of the variable reflect the process of the command. Example: COTx.
C_DC_NA_1	46	USINT	The variable reflects the value of the command that has been received from the 870 master and the status bits of the variable reflect the process of the command. Example: COTx. 0 or 1, with DPI/DCS Mapping
C_DC_TA_1	59	USINT	The variable reflects the value of the command that has been received from the 870 master and the status bits of the variable reflect the process of the command. Example: COTx. 0 or 1, with DPI/DCS Mapping
C_RC_NA_1	47	USINT	The variable reflects the value of the command that has been received from the 870 master and the status bits of the variable reflect the process of the command. Example: COTx. 0 or 1, with DPI/DCS Mapping
C_RC_TA_1	60	USINT	The variable reflects the value of the command that has been received from the 870 master and the status bits of the variable reflect the process of the command. Example: COTx.

			0 or 1, with DPI/DCS Mapping
C_SE_NA_1	48	REAL	The variable reflects the value of the command that has been received from the 870 master and the status bits of the variable reflect the process of the command. Example: COTx.
C_SE_TA_1	61	REAL	The variable reflects the value of the command that has been received from the 870 master and the status bits of the variable reflect the process of the command. Example: COTx.
C_SE_NB_1	49	INT	The variable reflects the value of the command that has been received from the 870 master and the status bits of the variable reflect the process of the command. Example: COTx.
C_SE_TB_1	62	INT	The variable reflects the value of the command that has been received from the 870 master and the status bits of the variable reflect the process of the command. Example: COTx.
C_SE_NC_1	50	REAL	The variable reflects the value of the command that has been received from the 870 master and the status bits of the variable reflect the process of the command. Example: COTx.
C_SE_TC_1	63	REAL	The variable reflects the value of the command that has been received from the 870 master and the status bits of the variable reflect the process of the command. Example: COTx.
C_BO_NA_1	51	UDINT	The variable reflects the value of the command that has been received from the 870 master and the status bits of the variable reflect the process of the command. Example: COTx.
C_BO_TA_1	64	UDINT	The variable reflects the value of the command that has been received from the 870 master and the status bits of the variable reflect the process of the command. Example: COTx.
C_IC_NA_1	100	-	The command is executed automatically in the background. All ASDUs envisaged by the IEC 60870 standard are sent to the 870 master
C_CI_NA_1	101	-	The command is executed automatically in the background. All ASDUs envisaged by the IEC 60870 standard are sent to the 870 master
C_RD_NA_1	102	-	The command is executed automatically in the background. All ASDUs envisaged by the IEC 60870 standard are sent to the 870 master

C_CS_NA_1	103	BOOL	Enable variable of the time synchronization
C_TS_NA_1	104	–	The command is executed automatically in the background. All ASDUs envisaged by the IEC 60870 standard are sent to the 870 master
C_TS_TA_1	107	–	The command is executed automatically in the background. All ASDUs envisaged by the IEC 60870 standard are sent to the 870 master
C_RP_NA_1	105	USINT	The variable reflects the value QRP of the command that has been received from the 870 master.
F_DR_TA_1	126	USINT	The variable of the triggering of the spontaneous directory transfer (on page 221)

AMENDMENT OF IO VALUES

It must be possible to link an IO (with a certain type ID) to a variable whose value exceeds the value range of the type ID of the IO.

Example: An IO with Type ID = 1 - single-point information was linked to a UDINT variable */stVal[ST] from the IEC850 driver.

In this case, the Process Gateway will amend the values for IO and also set the quality bits in ASDU, as envisaged in the IEC 60870 standard.

ASDU type ID	zenon variable value	Value and quality in ASDU
M_SP single-point	< 0 > 1	0 + IV 1 + IV
M_DP double-point	< 0 > 3	11b + IV 11b + IV
M_ST step position	< -64 > 255	x40 + OV xFF + OV
M_BO bitstring 32-bits	< 0 > xFFFFFFFF	0 + OV xFFFF FFFF + OV
M_ME_xA normalized	< -1.0 ≥ 1.0	x8000 + OV x7FFF + OV
M_ME_xB scaled	< -2 ¹⁵ > 2 ¹⁵ -1	x8000 + OV x7FFF + OV
M_ME_xC short floating	< -MAX Float > MAX Float	-MAX Float + OV MAX Float + OV
M_IT integrated totals	< -2 ³¹ > 2 ³² -1	x8000 0000 + IV x7FFF FFFF + IV



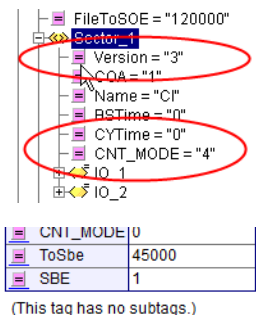
Information

If the zenon variable currently has the status bit `INVALID` or the status bit `OV_870` set, the value of the IO is also sent with `Quality IV` or `Quality OV`.

You can find further information in relation to this in the "Mapping of status bits in message direction (on page 215)" chapter.

Sector

XML FIELDS



CNT_MODE	0
ToSbe	45000
SBE	1

(This tag has no subtags.)

Parameter	Description
Version	The CNT_MODE field is expected from version 3 onwards.
CNT_MODE	Configured mode of integrated totals IOs.
Sector	<p>The new items ToSBe and ob SBE are added for the sector:</p> <ul style="list-style-type: none"> ▶ ToSBe: Timeout [ms] for Select before execute: 1000 - 65535000 ▶ SBE: 0 => not required 1 => necessary for execute

CONFIGURED MODE

Parameter	Description
0	With general interrogation. (Default)
1	Mode A
2	Mode B
3	Mode C
4	Mode D

ERROR CODES

Code	Description
13	Advise for integrated totals fail.
14	Queue overrun. Integrated totals values lost.
16	Advise failed for command variable
17	<p>List of value changes for command variables with SBE too long. Value change has been lost.</p> <p>2048 value changes are buffered as a minimum, or four times the number of active Select Before Execute (SBE) - depending on which value is greater.</p>
572	Counter interrogation. Unsupported QCC.
573	Counter interrogation. Unexpected IOA.
574	Counter interrogation. Unsupported COT.
575	Buffer for integrated totals interrogation data too small.

WARNINGS

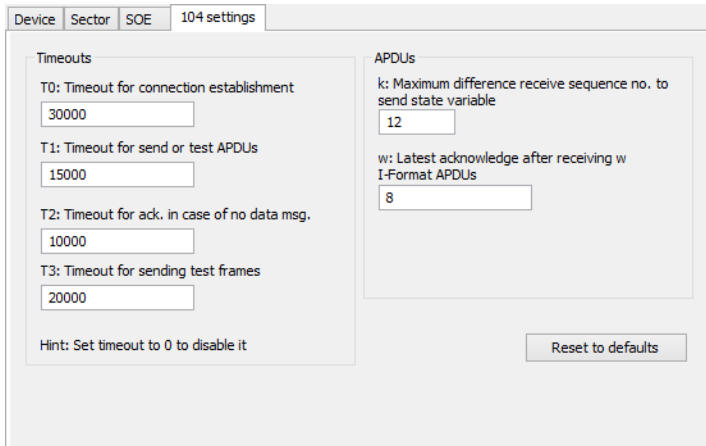
Code	Description
1011	QCC FRZ is only partly supported.
1012	Counter interrogation already active.
1014	Activation confirmation already confirmed.

ERROR MESSAGE

Message	Description
Unsupported qualifier of counter interrogation. <command>	Unknown QCC. RQT is only supported with 5. For C_CI_NA_1 (Counter interrogation) it is the case that RQT != 5 is treated the same as RQT = 5.
Counter interrogation processed without reset. < command>	Warning. Reset is not supported.
Can't advise IT data point. Device:(<device>)<device> Sector:(<COA>)<Sektorname> adv:0x<Advise-ID>	Data points for integrated totals could not be requested.
IT Values lost, queue full! Device:(<device>)<device> Sector:(<COA>)<Sektorname> <Wertänderung>	The buffer for the value changes of IT IOs is full. The value change is lost.

10.1.5 104 settings

In the **104 settings** tab, you set the timeout times of the Link Layers and the number of APDUs . The settings should be the same for slave and master. If a master overwrites a timeout or the number of frames to be confirmed, the slave closes the connection.



The screenshot shows the '104 settings' tab in the zenon software. It contains two main sections: 'Timeouts' and 'APDUs'. The 'Timeouts' section has four input fields: T0 (30000), T1 (15000), T2 (10000), and T3 (20000). The 'APDUs' section has two input fields: k (12) and w (8). A 'Reset to defaults' button is located at the bottom right. A hint at the bottom left states: 'Hint: Set timeout to 0 to disable it'.

Section	Parameter	Value
Timeouts	T0: Timeout for connection establishment	30000
	T1: Timeout for send or test APDUs	15000
	T2: Timeout for ack. in case of no data msg.	10000
	T3: Timeout for sending test frames	20000
APDUs	k: Maximum difference receive sequence no. to send state variable	12
	w: Latest acknowledge after receiving w I-Format APDUs	8

Note: The input fields of this tab are only available if 870-104 has been selected as a protocol.

Parameter	Description
Timeouts	Timeout settings of the connection
T0: time-out of connection establishment	<p>Timeout for establishing connection.</p> <p>Value range 0 - 4294967295</p> <p>Default: 30000</p> <p>There should also be at least one u-frame from the master in that time.</p>
T1: Time-out send or test APDUs	<p>Timeout for frame confirmation by the master.</p> <p>Value range: 0 - 4294967295</p> <p>Default: 15000</p>
T2: Time-out for ack. in case of no data msg	<p>Timeout, within which the master should confirm if no data is exchanged.</p> <p>Value range: 0 - 4294967295</p> <p>Default: 10000</p>
T3: Time-out for sending test frames	<p>Time after which a U-frame is sent to the master if no data is to be transferred.</p> <p>Value range: 0 - 4294967295</p> <p>Default: 20000</p>
APDUs	<i>Settings for the ISO 7816 standard: Application Protocol Data Unit (APDU).</i>
k: Maximum difference receive sequence nr to send state	<p>Number of maximum I-frames not yet confirmed by the master.</p> <p>Value range: 0 - 4294967295</p> <p>Default: 12</p>
w: Latest acknowledge after receiving w I-Format APDUs	<p>Number of I-frames received after a confirmation is sent.</p> <p>Value range: 0 - 4294967295</p> <p>Default: 8</p>

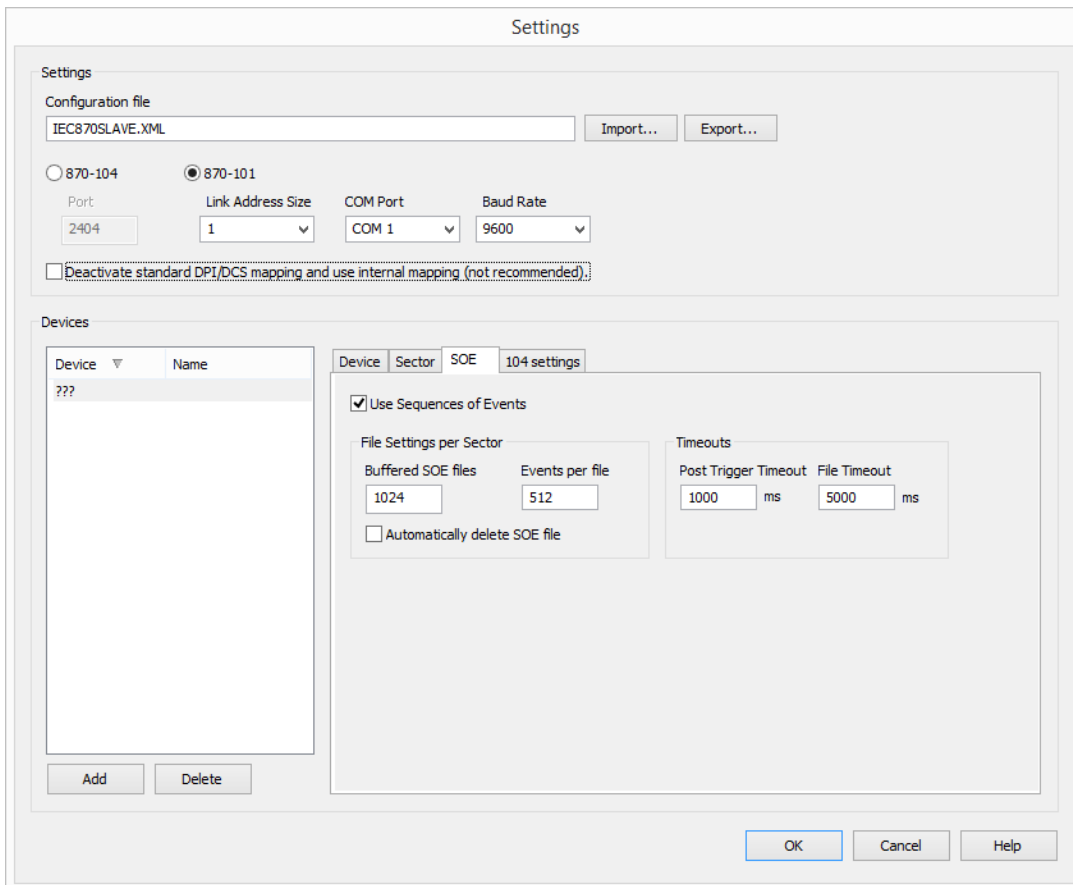


Information

The **Reset to default** button resets all input to the values prescribed by the IEC 60870-5-104 standard.

10.1.6 DPI / DCS mapping

It is possible to configure whether DPI/DCS (double-point information/double command state) should be transferred in accordance with the IEC 60870 standard or in the zenon internal display. The default is transfer between Process Gateway and an 870 master in accordance with the standard.



The screenshot shows the 'Settings' dialog box for the IEC870 Slave configuration. The dialog is divided into two main sections: 'Settings' and 'Devices'.

Settings Section:

- Configuration file:** A text box containing 'IEC870SLAVE.XML' with 'Import...' and 'Export...' buttons.
- Protocol Selection:** Two radio buttons: '870-104' (unselected) and '870-101' (selected).
- Port Configuration:**
 - Port:** A text box with '2404'.
 - Link Address Size:** A dropdown menu with '1' selected.
 - COM Port:** A dropdown menu with 'COM 1' selected.
 - Baud Rate:** A dropdown menu with '9600' selected.
- Mapping Option:** A checkbox labeled 'Deactivate standard DPI/DCS mapping and use internal mapping (not recommended)' which is currently unchecked.

Devices Section:

- Device List:** A table with columns 'Device' and 'Name'. It contains one entry '???'.
- Buttons:** 'Add' and 'Delete' buttons are located below the device list.
- Configuration Tabs:** 'Device', 'Sector', 'SOE', and '104 settings'. The 'SOE' tab is currently selected.
- SOE Tab Configuration:**
 - Use Sequences of Events:** A checked checkbox.
 - File Settings per Sector:**
 - Buffered SOE files:** A text box with '1024'.
 - Events per file:** A text box with '512'.
 - Automatically delete SOE file:** An unchecked checkbox.
 - Timeouts:**
 - Post Trigger Timeout:** A text box with '1000' followed by 'ms'.
 - File Timeout:** A text box with '5000' followed by 'ms'.

Bottom Buttons: 'OK', 'Cancel', and 'Help' buttons are located at the bottom right of the dialog.

Setting	Functionality
Deactivate standard	<p>A setting for compatibility when converting projects from very early versions of zenon or use with zenon drivers for which the Mapping of Double Point Values has also been deactivated.</p> <p>Default: Inactive.</p> <p>Inactive: Values for DPI/DCS (and RCS) are mapped as follows, which also guarantees the correct behavior for the Command Processing module as well as for communication with an 870 master:</p> <ul style="list-style-type: none"> ▶ zenon value 0 <-> DPI/DCS/RCS value 1 (= OFF/LOWER) ▶ zenon value 1 <-> DPI/DCS/RCS value 2 (= ON/HIGHER) ▶ zenon value 2 -> DPI value 0 (intermediate) ▶ zenon value 3 -> DPI value 3 (failure/error) <p>DPI is "double-point information" IOs; DCS is "double command" IOs; RCS is "regulating step command"; Mapping from zenon version 8.00</p> <p>Active: Mapping is deactivated. Values for DPI/DCS/RCS correspond directly to the values of zenon. Not recommended.</p> <p>Note: For compatibility reasons, this setting is also activated if an XML file is read with the value 1 in the <code>Version</code> field in the configuration of the Process Gateway.</p>

10.1.7 Configuration via XML file

The configuration of the IEC870SI is stored in an XML file. The active XML file is read when the gateway is started. The active configuration is stated in the INI file (on page 19), for example:

[IEC 870]

SETTINGS=IEC870SLAVE.XML

You can choose any name. The path is the same as in the INI file.

Note: You can import and export different configurations with the buttons '**Export**' and '**Import**' in the main window.

STRUCTURE OF THE XML FILE

The fields in the XML file correspond to settings that have been approved in the respective configuration dialogs.

The values and structure of the XML file are verified when the file is read. Erroneous entries trigger an error message that is also logged in the LOG - **Diagnosis Viewer**.

**Attention**

XML entries are case sensitive. Pay attention to use correct upper and lower cases.

Name	Property
process_gateway_IEC870Slave	The root node must have this name. Otherwise, the reading process will stop with an error.
Version	Contains the current version of attributes and nodes.
Is101active	Data Layer Selection: <ul style="list-style-type: none"> ▶ 0: 870-104 ▶ 1: 870-101 unbalanced ▶ 2: 870-101 balanced
DeActDPIStd	DPI /DCS compliant (IEC60870 compliant) data transfer or internal control system format

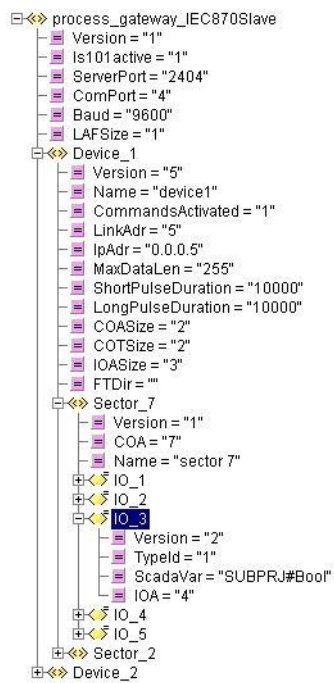
Device_x	<p>Nodes for the device configuration. When saving, X is replaced by a consecutive number, starting with 1.</p> <p>Note: No particular node naming is required during import. A Device is created for each node.</p>
Version	From 6 describes an XML file with the fields for SOE support
LinkAdr	In protocol 101, the Link Address is set here. It must be unique for all devices. For protocol 104, this entry has no meaning.
IpAdr	In protocol 104, the IP address of the Master is entered here. It must be unique for all devices. For protocol 101, this entry has no meaning.
EnableSOE	<ul style="list-style-type: none"> ▶ 0 = inactive ▶ 1 = SOE active
MaxBufferedFilesSOE	The maximum number of buffered SOE files.
AutodeleteFilesSOE	<ul style="list-style-type: none"> ▶ 0 = no Autodelete ▶ 1 = autodelete if the maximum number of the SOE files has been reached
MaxEventsPerFileSOE	Maximum number of ASDU per SOE file
PostTrgToSOE	Post Trigger Timeout in [ms]
FileToSOE	File Timeout in [ms]

Sector_x	<p>Nodes for the configuration of a sector. X is replaced by the COA when saving.</p> <p>Note: No particular node naming is required during import. A sector is created for every node.</p>
Version	From 4 marks an XML file with the fields for Select Routing support.
COA	<p>Common Address of ASDU (Common Object Address)</p> <p>Must be unique for every device per sector.</p>
BSTime	<p>Periods for Background Scan.</p> <p>Unit of measurement: ms.</p> <p>0 deactivates the background scan.</p> <p>Default: 0</p>
CYTime	<p>Periods for cyclical data transfer</p> <p>Unit of measurement: ms.</p> <p>0 deactivates the cyclical transfer.</p> <p>Default: 0</p> <p>Available in version 2 and above</p>

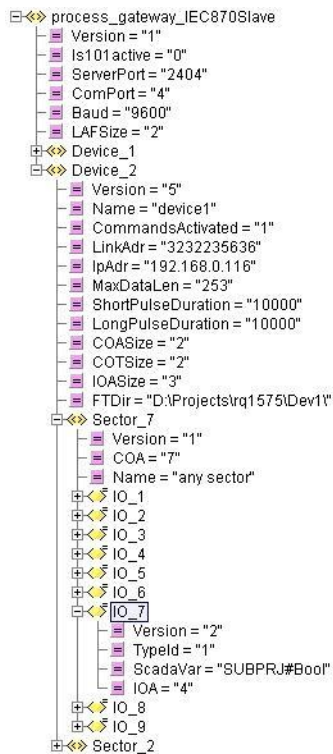
IO_x	<p>Nodes for the configuration of an Information Object. When saving, X is replaced by a consecutive number, starting with 1.</p> <p>Note: No particular node naming is required during import. A IO is created for each node.</p>
Version	<ul style="list-style-type: none"> ▶ From 4 marks an XML file with the fields for SOE support; ▶ From 5 marks an XML file with the fields for support of hysteresis;
TypeId	<p>Type Identification</p> <p>When loading, a check is carried out to see whether the Type ID is possible with the data type of the given variables.</p>
ScadaVar	<p>Name of the zenon variable, syntax:</p> <p>[ScadaProjectName#]ScadaVariableName.</p> <p>Variables without a project reference are searched for in the default project (INI file - [GENERAL] section: PROJECT=).</p> <p>When the XML file is loaded, the existence of the specified variable in the project is verified.</p>
IOA	<p>Information Object Address</p> <p>In combination with the TypeId, this must be unique per sector and the direction (controlling/monitoring) must be unique.</p>
KindOfTransfer	<p>States the type of data transfer for IO in monitoring direction .</p> <ul style="list-style-type: none"> ▶ 0 spontaneous (COT=3) ▶ 1 Background scan (COT=2) ▶ 2 periodic/cyclical (COT=1) ▶ 3 off; no transfer to the 870 master ▶ 4 spontaneous transfer with buffering in the event of a communication failure <p>Default: 0</p> <p>Available in version 3 and above</p>
TrgSOE	<ul style="list-style-type: none"> ▶ 0 = inactive ▶ 1 = the IO acts as a trigger for SOE
ExDetectionSOE	<p>Determines when a change to the value or status bit of the IO can trigger a write to the SOE file.</p> <ul style="list-style-type: none"> ▶ 0 SOE active ▶ 1 Saves all changes. ▶ 2 Raising edges ▶ 3 Falling edges ▶ 4 Raising and falling edges

HysteresisNegative	Value of the negative hysteresis.
HysteresisPositive	Value of the positive hysteresis.

Example configuration for the protocol 870-101:



Example configuration for the protocol 870-104:



.CSV export and import

Configurations of the Process Gateways can be exported into a text file or imported into Process Gateway as a text file.

The first line in the CSV file must be a header with the names of the properties. The individual properties must be separated with a TAB.

The following information of the IO must be saved per line of the CSV file:

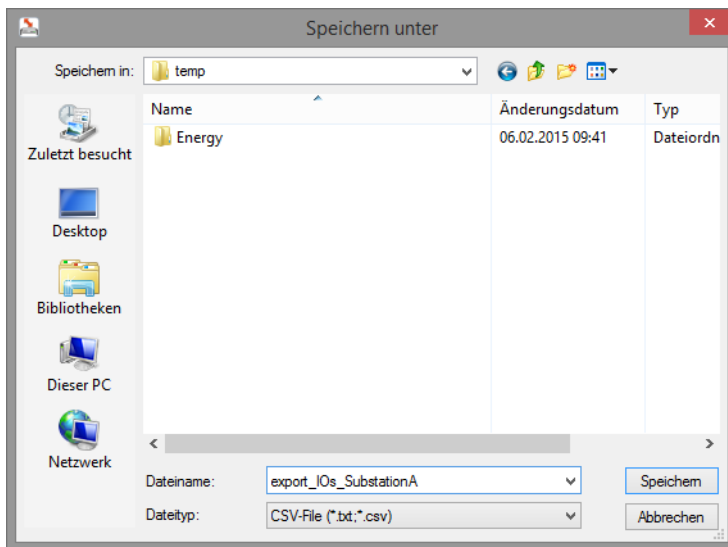
- ▶ LinkAdr or IpAdr of the device
- ▶ COA of the sector
- ▶ All settings of the IO

New IOs are typically created in the Process Gateway during import.

In addition, the `KindOfTransfer` property can be used during import to update or delete pre-existing IOs of the Process Gateway configuration.

EXPORT

Carry out the following steps in order to export a Process Gateway configuration:



1. Click on the **Export...** button in the configuration dialog.
The file selection dialog is opened.
2. Select the `CSV-File` entry in the **File Type** drop-down list.
3. Name the export file in the **Filename** input field
4. Click on **Save**.
The file is exported and saved at the selected save location.

IMPORT

The configuration of the Process Gateway can be created with a .CSV file or supplemented with a .CSV file. The .CSV file can come from another process gateway or be created with an external program (such as MS Excel).

Carry out the following steps to import a .CSV file into the configuration of the Process Gateway:

1. Click on the **Import...** button in the configuration dialog.
The file selection dialog is opened.
2. Select the `CSV-File` entry in the **File Type** drop-down list.
3. Select the file.
4. Click on **Open**.
The file is loaded and the configuration of the Process Gateway is updated.

RULES FOR IMPORT

The following rules are applicable for import and amendment:

1. `LinkAdr` or `IpAdr`:

- ▶ The import of a `LinkAdr` is only permitted if the Process Gateway has already been configured for 870-101.
- ▶ The import of a `IpAdr` is only permitted if the Process Gateway has already been configured for 870-104.

The following is applicable in the event of a conflict between protocol and addressing:

- If no device has yet been created in the Process Gateway, the import process amends the protocol setting accordingly and creates a device in the Process Gateway. This device is configured with default settings.
- If there is already a device present in the Process Gateway (for a different protocol), the import is canceled with an error message.

1. Sector

- ▶ If no sector has yet been configured for the COA of the import file to be imported, the import process creates a new sector.
- ▶ The IO is then imported into this sector.

1. IO

The `KindOfTransfer` column determines the editing layer of the IO:

- ▶ 0...4 - normal value range for `KindOfTransfer`
Creates new IO
- ▶ 100...104
Deletes the IO from the sector that has precisely `IOA` and `TypeID`
- ▶ 300...304
Deletes the IO from the sector that is linked to `ScadaVar`.
- ▶ 200...204
Updates the IO that has precisely `IOA` and its `TypeID` is in the appropriate area.
Areas are applicable:
 - 1...37
Monitoring Direction
 - 45...126
Controlling Direction
 - 0
for internal `TypeIDs`
- ▶ 400...404
Updates IO that is linked to `ScadaVar`.

After this editing step, `KindOfTransfer` is then converted to the normal value range.



Example

In the Process Gateway, an **IO** with `COA=1`, `IOA=1` and `TypeID=1` was configured for spontaneous transfer, and linked to the variable with the name `Var1`.

The import file contains, for variable `Var1`, a line with `COA=1`, `IOA=100`, `TypeID=30` and `KindOfTransfer=400`. As a result, for variable `Var1`, the `TypeID` of the **IO** is changed from 1 to 30 and `IOA` is changed from 1 to 100.

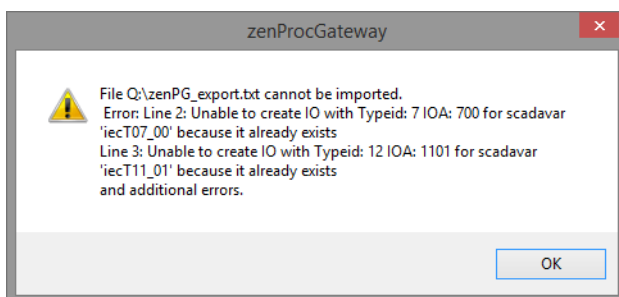
The **Data Transfer** remains spontaneous.

ERROR TREATMENT

If an error is detected during the import process, an entry is created in the LOG file. In addition, a corresponding warning dialog is shown in the Process Gateway at the end of the import.

Possible error causes:

- ▶ An IO is not yet present in a sector that has already been configured, but envisaged for updating in the file with `KindOfTransfer` .
- ▶ The import does not allow more than one IO with a certain `IOA` to be created in the respective `TypeID` area.
- ▶ The creation of IOs for variables that are not present in the project or that have a non-numerical data type is considered an error.



The header is applicable as a line with number 1.

10.2 Sequence of Events (SOE)

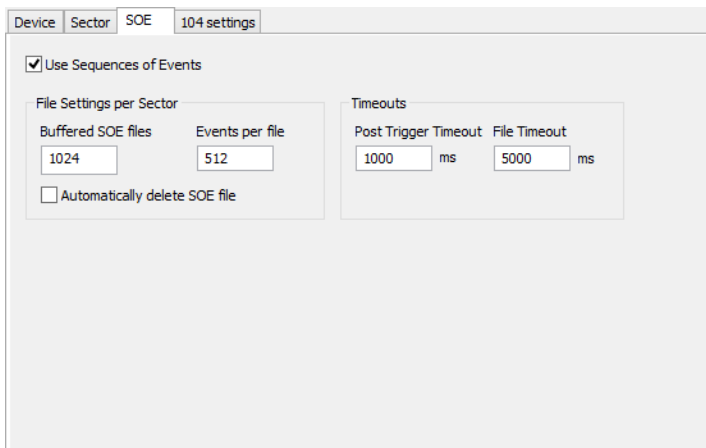
The **Process Gateway IEC870 Slave** supports the automatic recording and transfer of SOE files - in accordance with procedures defined in IEC 60870-5-101, Section 7.4.11.3 "Transmission of sequences of events" (in a section of a data file).

In the sector, at least one IO must have the **File Trigger** option activated, otherwise no SOE files are created.

Note: You can find the **File Trigger** option in the **Sector** tab - **Information Objects** (on page 172), in the **Sequence of Events (SOE)** area.

The SOE files are saved in the folder for file transfer. A folder for file transfer must therefore be configured in **Device (Directory for File Transfer option)**.

SETTINGS IN THE SOE TAB



Parameter	Description
Use sequences of events	<p>If active, the SOE for the selected device is activated. If inactive, all SOE settings to the IOs are ignored.</p> <p>Default: inactive</p> <p>Note: The IEC870 slave only saves, in SOE files, events of the IOs for which the SOE has been activated in the Sector - Information Objects (on page 172) tab.</p>

FILE SETTINGS PER SECTOR

Parameter	Description
Buffered SOE Files	<p>If the number set for SOE files in the folder is reached for the sector, a spontaneous directory transfer (on page 221) (type ID <code>F_DT_TA_1</code>) is triggered. If the number of files in the folder is doubled, the oldest file will</p>

	<p>start to be deleted (ring buffer functionality).</p> <p>Value range 1 to 1024.</p> <p>Default: 1024</p> <p>Attention: A directory - but not file transfer - is triggered automatically. The SOE files must be requested by the 870 master.</p> <p>Note: The current status of the SOE can be evaluated using variables that are linked to the internal type T00 (on page 210).</p>
Events per File	<p>Number of value changes that are written in a file before it is closed.</p> <p>Value range: 1 to 4096.</p> <p>Default: 512</p>
Automatically delete SOE file	<p>If active, the reaction is that this is then automatically closed for type ID "F_AF_NA_1" (confirmation for file) for an SOE file.</p>

TIMEOUTS

Parameter	Description
Post trigger timeout	<p>If no ASDU is written to the file within this period of time, this is then closed. The time period restarts with each ASDU written.</p> <p>Value range: 100ms to 120000ms</p> <p>Default: 1000ms</p>
File timeout	<p>Maximum time for which the file remains open (regardless of the number of events).</p> <p>Value range: 100ms to 120000ms</p> <p>Default: 5000ms</p>

FIRST VALUE

The exchange of data between IEC870SI and zenon Runtime for SOE starts immediately when the gateway has finished installing. The first value that is initialized does not trigger an Event and is thus also not entered into the file. The same applies if Runtime is ready again to exchange data after reloading.

The exchange of data for data in Monitoring direction only starts when a master has connected and triggered a general interrogation (GI) (on page 205).

FILE TRANSFER

As part of the SOE functionality for transparent (1), event (3) as **NOF** is also supported as **NOF**.

There is no automatic File transfer if the maximum number of SOE files to be buffered has been exceeded.

In the current version, the gateway supports spontaneous directory transfer, but not file transfer activated by the slave. The SOE files must be requested by the 870 master.

Note: the current status of the SOE can be evaluated using variables that are linked to internal type T00 (on page 210).

DIRECTORY TRANSFER

In accordance with the IEC 60870 standard, an automatic spontaneous directory transfer (on page 221) is activated for the following reasons with existing SOE files:

- ▶ 24 hours after the last automatic directory transfer.
- ▶ if the master connects.
- ▶ if the maximum number of files to be buffered is exceeded.

10.2.1 Warnings

Warning message	Meaning
Sequences of events for device <Devicename> sector <Sectorname> are activated but no trigger defined	The device has SOE activated, on the server there is an IO with SOE active but the trigger was not activated for any IO. No file is created without a trigger. At least one IO must have the trigger activated for the sector.
SOE value change without trigger and file is closed. Value lost! Device:(<Device ID>)<Device-Name> Sector:(<COA>)<Sektorname> <Wertinformation>	A value change was not written because the SOE file is not open. Code:1010

10.2.2 Error message

Error message	Meaning
Sequence of events active but no directory for the file transfer defined!	SOE files are archived in the file transfer folder. Therefore a folder for file transfer must be present.
SOE Values lost, queue full! Device: (<Device ID>) <Device-Name> Sector: (<COA>) <Sektorname> <Wertinformation>	for IOs that have a SOE activated, data is exchanged to the stack via a queue. This can assume a value of 2048 or 4 * the number of SOE IO's in sector entries. If this number is exceeded, this error message is created and the value change is lost. Error code: 6
Can't create subdirectory for file transfer. Device: (<Device ID>) <Devicename> Sector: (<COA>) <Sektorname> system error: <Systemfehlercode>	The subfolder for the sector is automatically created for SOE in the folder for file transfers. This error shows that this folder could not be created. SOE then indicates that this sector is NOT available. Error code: 10
Systemfehlercode	Contains the error code returned from the operating system.
Can't advise SOE data point. Device: (<Device ID>) <Devicename> Sector: (<COA>) <Sektorname> adv: 0x<Advise-ID>	Data point request for SOE IOs failed. SOE then indicates that this sector is NOT available. Error code: 11
Write asdu to file <Dateiname> fail! Device: (<Device ID>) <Devicename> Sector: (<COA>) <Sektorname> <Value> system error: <Systemfehlercode>	ASDU write to SOE file failed. Error code: 8
Illegal file name <Dateiname> detected! Device: (<Device ID>) <Devicename> Sector: (<COA>) <Sektorname>	File names for the file transfer have a defined format <IOA>.<NOF>. The file does not adhere to this convention. Error code: 12
Auto delete for file failed! System error: <Systemfehlercode> <Dateitransferinformation>	Error code: 9
Dateitransferinformation	Contains the information that identifies the file transfer.

10.3 Communication

This chapter explains details on communication in the **Process Gateway**:



Information

If an **ASDU** with unknown **Type-ID** is received by the **Process Gateway**, a negative response of **COT 44** is sent to the 870 master.

10.3.1 Start

It loads the INI file when the **Process Gateway** is started. The **Process Gateway** then attempts to read its active configuration from the XML file - entered in the INI file. If this is not possible, an error message is displayed and the configuration dialog is opened.

If the configuration dialog is not left by clicking on the **OK** button, the **Process Gateway** is closed.



Attention

*No configuration dialog is called up if the **Process Gateway** is started invisibly. The **Process Gateway** will automatically quit if its configuration is invalid or missing.*

After the IEC870 slave has successfully read its configuration from an XML file, it expects the establishment of a connection from a master. The IEC 60870 standard defines that a session between a master and slave must always start with a general query (**GI - C_IC_NA_1**). This is why the receipt of the GI from the master is a requirement for spontaneous communication. First the general query triggers the establishment of the connection to zenon Runtime in the IEC870 slave and the request for the current values of the variables for IOs.

Note: if a master opens the connection to the link layer but does not send a general query, you can attempt to provoke the GI by sending an **ASDU T70 - End of Initialization** (on page 208).

10.3.2 General interrogation

The general query (GI) triggers an establishment of the connection to zenon Runtime in the IEC870 slave. The receipt of the **GI - C_IC_NA_1** - from a master is the requirement for spontaneous communication, **Background scan** or **Periodic data transfer**. As long as the IEC870 slave has not received the GI, it does not send any values to the master.



Attention

In accordance with the IEC60870 standard, a master must start communication with a general query. For this reason, the slave closes the connection if the master has not sent a GI in the envisaged time (T_0).

Note: if a master opens the connection to the link layer but does not send a general query, you can attempt to provoke the GI by sending an ASDU T70 - End of Initialization (on page 208).

After receipt of the GI, the connection to zenon Runtime is set up in sectors (COA). If the connection cannot be established in Runtime, there is a negative confirmation of the general query (COT_actcon + PN). The query that is already active for the sector is ended and a warning is logged.

A GI for a sector that does not exist in the slave (unknown COA) is confirmed in the negative (COT=46 + PN) and the connection to the master is disconnected. This is an indicator for a configuration error in the system.

Other system configuration errors are handled with more tolerance. The receipt of invalid values is logged in the LOG file. The following is applicable in this case:

- COT != 6 Is responded to with COT 45 and negatively. The connection remains intact.
- QOI != 20 Is confirmed positively and nevertheless triggers Station interrogation. This corresponds to the behavior the same as for QOI = 20.
- IOA != 0 Is handled the same as IOA=0.

For general queries, the slave supports the global address COA=0xFF (FF) . You can find detailed information on this in the Broadcast (on page 224) chapter. A master can thus query several sectors with just one GI. This is an alternative envisaged by the standard, which the master can use. Otherwise a master must query the sectors consecutively (one after the other).

If the master sends a further GI whilst the current one has not yet been concluded, the ongoing GI is canceled and the new one is started. Overlapping GIs are logged as a warning. This is an indication of non-standard-compliant behavior with a master.

Value changes for IOs whose value has already been reported to the master as part of the current GI are buffered as long as the GI - due to other IOs is still running in the same sector (COA). If a sector has many IOs whose values change often and the connection to the master is comparatively slow, the buffer can grow more quickly than the master takes the values. This can happen if, for example, the **Baud Rate** is small, the **Max.user data** has been limited or the master reacts slowly. In such cases, the distribution of the IOs over several sectors can be advisable.



Attention

If the internal buffer is overfilled with value changes before the GI could be completed, the current connection to the master is disconnected.

You can find further information on this in the Transfer of the value changes (on page 209) chapter.

The connection to Runtime cannot be established whilst a project is reloaded (on page 209) in Runtime. The general request will fail in this case.

The GI and the buffering of the value changes are independent for different **Devices** (for different masters). A master can deactivate the ongoing general query by ASDU<100> with `COT_deact(8)`.

If the connection to Runtime has been established, but Runtime does not provide values for some IOs, the slave can confirm the GI but not close it (`noCOT_actterm`). This is applicable for the IOs that are to be reported to the master (= GI) on the basis of the current query. Often, the error that no values can be provided is due to a configuration error.

Example: a write-only zenon variable (an output or command) is in the slave with an IO linked to the **Type ID** of a message (T01-T37).

The slave logs the information using **IOA** and **COA** addresses of the affected IOs until it detects the required values. This logging is carried out cyclically. This information is logged with level `DEEPDEBUG` in the LOG file and can be read with the `Diagnosis Viewer`.

Text: "Information object <IOA> missing value. Device: <IP> Sector: <COA>".

10.3.3 End of connection and resets

The reset in the slave is triggered if a master connects:

- ▶ 870-101: All commands will be rejected until the function code for reset (0) has been received. The code causes a reset in the link layer and the application layer.
- ▶ 870-104: As a result of the connection being disconnected, a master can detect the interruption immediately and starts the initialization process again.

ENDING COMMUNICATION

In general, communication is ended if:

- ▶ The Process Gateway application is ended
- ▶ Changes to the configuration of the slave are activated

- ▶ Runtime reloads
- ▶ the master has established the connection but no general query was sent
- ▶ The list of the value changes that have not been sent for the sector has overflowed
- ▶ A non-supported command has been received (command is answered in the negative and the connection is ended for security reasons).
- ▶ A command with a non-supported address is received (command is answered in the negative and the connection is ended for security reasons).

'Reset process command' - T105 - request

The slave can inform zenon Runtime of the receipt of a `C_RP_NA_1` command. To do this:

- ▶ create an IO of type `T105` with `IOA 0`.

If a master sends the command `C_RP_NA_1`, the value of the variable is first set to 0 and then to the value of `QRP`.

The slave confirms receipt of the command to the master and changes the value of the corresponding `T105` variable for Runtime but otherwise the command has no effect in the slave. The optional sending of the 'End of initialization' (`T70`) is not executed automatically.

'End of Initialization' - T70 - send

From the process control system, it is possible to instigate an `ASDU<70> - M_EI_NA_1` ('End of initialization') being sent to the master. To do this:

- ▶ Create a type `T70` IO with `IOA 0`.
- ▶ and link to an internal variable with `UINT` data type

The transfer is triggered if the current value of the variable is 126 and this is amended to 127. As COI (Cause of initialization), 2 (remote reset) is used.

Note: some masters only send the general interrogation (GI) if they have received an End of initialization.

CONFIGURATION TIP

In systems where the 870 master expects an End of initialization in order to send the general interrogation, you can change the value of the variable that has linked the IO to `T70` and `IOA 0` between 126 and 127 if the other variable with `T00` and `IOA 1` (Master connected) (on page 210) changes the value. To do this, create two **Write set value** functions for the `T70` variable and link these functions in the **limit values** of the `T00 IOA 1` variable:

- When the `T00 IOA 1` variable has a value of 0 - set 126 to `T70` variable;

- And when the value is 1 - set 127.

10.3.4 Transfer of the value changes

The IEC870 slave receives, from zenon Runtime, all value changes of the requested variables. Because, as a result of this, the list of process information that has not yet been transferred to the master can become very large, the list of outstanding value changes is limited.

Limit: at least 2048 values or 4 x number of IOs per sector.

If a master receives the values more slowly than they are generated in the process control system, the list can exceed this maximum. The slave can then no longer buffer the value changes that have not yet been transferred. The slave therefore terminates the connection to the master. The master is thus requested to obtain the next connection with a general query, so that it gets current, only recent variable values.



Attention

A reset of the link layer will be forced every time there is a possibility of data loss (lost value changes) .

The Master must be able to recognize a communication interrupt, so he can react with a new general request. This is why no process commands will be executed as long as the master has not instigated a reset of the link.

10.3.5 Reloading Runtime

If one of the projects is reloaded in zenon Runtime, the connection to all projects is established and the connections to the master will be disconnected. For each `Device` in the IEC870 slave, a reset of the link layer is triggered. Accordingly, the rules for the start of communication are applicable, for example a general interrogation is expected.

Attention: While the reload is active, you cannot establish a connection to the Runtime. It is thus also not possible to close the general interrogation positively.

Note: After Runtime is reloaded, you can also - with the variable linked to `T00 IOA 8` - trigger reloading of the XML configuration of the Process Gateway.

10.3.6 Data priorities

The driver supports class 1 and class 2 files.

Spontaneous communication, **background scan** and **periodic data transfer** belong to class 1.

The whole file transfer is treated as class 2. This makes sure that value changes will be transferred to the Master even during a file transfer.

10.3.7 Internal type T00 (status variable)

The internal type **T00** is for the control and monitoring of the behavior of the IEC870 slave Process Gateway. The T00 variables are only communicated between the slave and the process control system, not to the 870 master.

The **IOs** with **T00** should be linked to numerical (**UINT**, **DWORD** etc.) zenon variables, from the **internal driver** for example.

IOA 1 - MASTER CONNECTED

The variable provides the status information as to whether there is already a Link Layer connection to the master. In **Device**, in the desired sector (**COA**), create a **T00 IO** with the **IOA 1**. If a connection is established, the value will change from 0 to 1.

Note: If a master opens the connection to the Link Layer without a general interrogation (on page 205) being sent, the **Process Gateway** will close after the expiry of the **T0** timeout. In such cases, you should attempt to provoke the GI from the master, for example due to the sending of an ASDU T70 - End of Initialization (on page 208).

IOA 2 - REJECT OR IGNORE COMMANDS

The variable determines how the IEC870 slave should handle commands that the 870 master sends. It is possible to decide for each sector (**COA**) whether commands are executed in control direction. The commands are always enabled if no **IO** is configured.

Note: The master's commands can also be rejected in general using the **commands active** (on page 161) property.

The zenon variable is assigned to **IO** with **T00** and **IOA 2**. You can link the same variable in all sectors (**COA**). The variable must not have the status **INVALID**.

The meaning of the variable:

Value	Description
0	<p>Disabled</p> <p>The ASDUs with commands are rejected.</p> <ul style="list-style-type: none"> ▶ For rejected commands, a master receives <code>COT_actcon (7) + PN</code> (negative) as confirmation. ▶ Rejected commands generate a warning entry in the LOG file. ▶ zenon Runtime is not informed of commands. <p>Example:</p> <p>Depending on the local/remote mode of a Substation with Process Gateway to the remote 870 master.</p> <p>The remote 870 master can be informed of the current mode via an IO in monitoring direction (with <code>T30</code> for example) by means of the actual mode, and whilst the Substation is in local mode, the remote commands are not permitted.</p>
1	<p>Enabled</p> <p>Commands are approved.</p>
2	<p>Silent mode</p> <p>The ASDUs with commands are ignored.</p> <ul style="list-style-type: none"> ▶ No reaction to the ASDUs with commands; without warning in the log. ▶ zenon Runtime is not informed of commands. <p>Example:</p> <p>Redundant connection to an 870 master of two Process Gateways that have been started with the same configuration: on two computers in the zenon network or on a computer with 2 Ethernet cards or with different IP ports.</p> <p>A command from this master should only be forwarded to zenon Runtime via one of the Process Gateways.</p>
3.	reserved for future versions



Information

Attendant warning message:

Commands for sector deactivated, execution prohibited. <Command text>

The meaning: A command in control direction was received, but rejected due to a missing enable.

IOA 3 - SOE: FILES OVERFLOW

Status variable is set to 1 if the number of SOE files in the transfer directory exceeds the maximum number of buffered files.

IOA 4 - SOE: CURRENT FILE STATUS

Variable provides status information on the current or most recently processed SOE file:

0 - initialization. As long as no SOE file was written.

1 - SOE file is opened

2 - file was closed

3 - too many SOE files in the directory

4 - error when writing the SOE file

5 - errors when creating the SOE file

IOA 5 - STATUS OF FILE TRANSFER

Variable provides status information by means of file transfer in `Monitoring Direction` ("Standard Direction" - from the master to the slave).

0 - no transfer active

1 - section is selected

2 - section is requested

3 - waiting on section confirmation

4 - waiting on confirmation of file transfer

255 - error when transferring

IOA 6 - SOE: NUMBER OF FILES

Variable provides the number of SOE files present in the transfer directory.

IOA 7 - STATUS OF FILE TRANSFER IN REVERSE DIRECTION

Variable provides status information by means of file transfer in `Control Direction` ("Reverse Direction" - from the slave to the master).

The information is integrated in a `DWORD` as follows: $[\text{IOA of the transferred files}] * 256 + [\text{status information}]$.

Possible status information:

1 = transfer is ongoing

2 - Transfer is ongoing - the transfer that had not been completed beforehand is terminated by the transfer that has currently been started.

Status 3 is also written for the transfer that has been terminated. However because the new status is immediately written afterwards, it cannot be guaranteed in the case of zenon that this is visible in Runtime.

3 - Transfer was terminated due to an error.

4 - transfer was completed and the file is available on the drive.

IOA 8 - RELOADING OF THE XML CONFIGURATION

In the IEC870 slave process gateway, it is possible to link a zenon variable to the `IOA 8` with type `ID<0>` `Internal Status`.

If this linked variable has the value `1` in zenon Runtime, the configuration for the Process Gateway is read in again. After successful reading-in of the new configuration, communication is restarted in the IEC870 slave and the value of the variable is reset to `0`. If this process is unsuccessful, the Process Gateway continues to work with the configuration loaded when the program started.

Restart and reloading of the XML configuration

From zenon 8.00, the restart for the Process Gateway - IEC870 slave can be triggered by a linked variable. With this restart, the XML file (on page 191) is read in again with the configuration.

GENERAL

In the IEC870 slave process gateway, it is possible to link a zenon variable to the `IOA 8` with type `ID<0>` `Internal Status`.

If this linked variable has the value `1` in zenon Runtime, the configuration for the Process Gateway is read in again. After successful reading-in of the new configuration, communication is restarted in the IEC870 slave and the value of the variable is reset to `0`. If this process is unsuccessful, the Process Gateway continues to work with the configuration loaded when the program started.

ENGINEERING

Link an **Information Object** to an internal zenon variable for type identification `8 - <0>` `Internal Status`.

To do this, carry out the following steps in the Process Gateway:

1. Select the **Sector** tab in the **Settings** configuration dialog.
2. Select a configured `Device` from the **Devices** list.
3. Select a configured `Sector` from the **Sectors** list.
4. Create a new `Information Object`.
 - a) To do this, select a pre-configured object with `IOA 8` and click on the **Edit** button or click on the **New** button to create a new `Information Object`.
The **Information object setting** (on page 172) dialog is opened.
 - b) Configure the `IOA` with the value 8 in the **Information Object Address** property.
 - c) In the **Type Identification** drop-down list, select the `<0> Internal status entry`.
Note: This entry is already selected by default.
 - d) Click on the ... button to link a zenon variable.
The **Variable selection** dialog to select a variable is opened.
 - e) Select a variable from the list in the Variable area.
 - f) Confirm the configuration by clicking on the **OK** button.
 - g) Click on the OK button to close the IO settings dialog.



Information

This configuration is possible in every sector. This configuration is applicable for the complete Process Gateway, not for individual sectors.

Configuration tip:

configure the link in one sector only or always link the same zenon variable in different sectors.

BEHAVIOR IN ZENON RUNTIME

When Runtime is started:

- ▶ The current values of the configured variables are read.
- ▶ The value of the linked variable for `IOA 8 - <0> Internal Status` is always reset to 0.
Note: The XML file with the configuration is only read once, even if the variable had the value 1 when communication was started.

When Runtime is running:

- ▶ When the value of the linked variable changes to 1:
 - a) All current Process Gateway connections are closed.
 - b) The Process Gateway rereads the XML configuration file.

- c) The value of the IOA 8 variable is set to 0.

Note: if there is no longer an **Information Object** with IOA 8 in the new configuration, the value for this variable remains 1.

- If the reading of the new configuration fails, communication continues with the old (= previous) configuration data. The variable for IOA 8 - <0> Internal Status is not reset.

CONFIGURATION TIP

With a variable for IOA 8 - <0> Internal Status , you have the following possibilities for selection in zenon Runtime:

- ▶ **Monitoring of the correct start of the Process Gateway:**
If the internal IOA 8 variable has an initial value greater than 1 (99 for example), this variable can be used to check that the Process Gateway has started correctly. If this initial value remains unchanged, the Process Gateway has not been started correctly.
Possible cause of error: the configured variables do not exist or the IP port is assigned.
- ▶ **Reloading the configuration for the Process Gateway via Runtime**
After reloading the project in zenon Runtime, the Process Gateway XML configuration file is not automatically reread.
If variables that have also been configured in the Process Gateway are amended or recreated in the zenon Editor, you can also trigger the reading of a new XML configuration file in the Process Gateway after reloading the project in zenon Runtime. To do this, set the value 1 of the IOA 8 variable to 1 in Runtime.

10.3.8 Mapping of status bits in message direction

The IEC870 slave supports the sending of quality bits for the information objects. The quality bits of the IO are set - in ASDUs in 'montior direction' - using certain status bits of the linked zenon variable.

The quality bits are assigned to the status bits of the variables as follows:

Status bit in zenon	IEC60870 quality descriptor bits	Notes
INVALID	IV (invalid) NT (not topical)	The slave forwards the <code>INVALID</code> status bit in message direction as two quality bits. Note: The IV quality bit can also be set due to an IO value amendment.
ALT_VAL	SB (substituted)	Up to zenon version 7.10, it is mapped the same as the <code>OFF</code> bit.
OFF	IV (invalid)	
N_UPDATE (not updated in zenon network)	NT (not topical)	Example: The zenon Process Gateway is running on a network client. During a redundancy switching of the network server, variables are primarily marked with <code>N_UPDATE</code> . The values of the IOs are sent to the 870 master with the NT quality bit. After redundancy switching has been completed, the network client automatically connects to the current network server and the values are spontaneous again. The values of the IOs are sent to the 870 master without the NT quality bit.
BL_870	BL (blocked)	
SB_870	SB (substituted)	Like <code>ALT_VAL</code>
NT_870	NT (not topical)	
OV_870	OV (overflow)	OV_870 status bits that have already been set for the variables in zenon are sent to the master. The following is also applicable for normalized values: The value range in accordance with the IEC 60870 standard is $<-1 \dots 1-2^{-15}>$ with a precision of 4 decimal places. With zenon variables, this corresponds to a value range of -1.0 to 0.9999 . Outside of the range for normalized values, in ASDU the value of the IOs is amended and the OV quality bit is set. Note: The OV quality bit can also be set to another type ID due to an IO value

		amendment.
T_INVALID (Zeit ungültig)	Only in "BinaryTime 2a" (CP24Time2a/CP56Time2a) BS[24] - IV (invalid)	The slave forwards the T_INVALID status bit in the time stamp of the IOs with "time tag" only.



Information

Process Gateway can mark the values with IV or OV Quality if the value of the zenon variable exceeds the value range of the ASDU type.

You can find further information in relation to this in the Assignment of the data types (on page 179) chapter in the "Amendment of IO values" chapter.

10.3.9 Commands

A command received - T45...T64 - is set as the value for the zenon variables. In sectors without `Select routing`, a confirmation, which is always positive, is always sent to the master immediately - `COT_actcon` (7) and `COT_actterm` (10).

In such sectors, a `Select` is also immediately also confirmed automatically - `COT_actcon` + `SE` and the zenon variable change neither the value nor the status bit.

Note: For variables that are configured in the IEC870SI Process Gateway as command variables, if the **Write set value via VBA** property is active (**Logging in** group), each command is logged as a **write set value** via the API interface in the **CEL**.

Select routing

The `Select routing` can only be used in conjunction with a project configuration of the zenon **command input** module and is only available for the **Auto/remote command** action.

The Process Gateway only reacts to a command variable by carrying out a `Select Routing` if the variable is configured in zenon with active **Select Before Operate** configured.

If, for the zenon variables, the **Select Before Operate** property is inactive, the **Process Gateway** behaves in the same way as for sectors without **Select routing** and always responds to a `Select` positively.

OVERVIEW OF ZENON VARIABLE CONFIGURATION - PROCESS GATEWAY

zenon variable Select Before Operate property:	Process Gateway - Sector option: Select Routing	Behavior in the Process Gateway
Active	Active	A Select is required. An Execute without Select is rejected. A Select is forwarded to the command input .
Active	Inactive	A Select is automatically responded to positively. The command input is not taken into account when executing the command. An Execute is always forwarded to the process driver directly.
Inactive	Active	

PROCEDURE

If, for the sector addressed, the `Select routing` option has been activated and **Select Before Operate** is activated for the zenon command variable, then there is a wait until a `Select` is received until the status `SE_870` is returned with `COT_actcon` (7) and possibly with `N_CONF` (corresponds to `PN`) via the command variable of zenon.

Only once this has happened is a confirmation sent to the master (for example a Control Center) triggered. All others for this IO are rejected as long as a selection via `Select` is taking place.

TIMEOUT, CANCEL AND MISSING SELECT

The following rules apply for the different possibilities for responding to a `Select routing`:

- ▶ If `Select routing` is configured for a sector, one `Select Timeout` per IO is started for each `Select`.
- ▶ The slave checks whether the `Timeout` has expired and then sets the variable to `SE`, `COT_actterm` (10) and `PN`.
- ▶ If a `Cancel` - `COT_deact` (8) is received by the master for an ongoing `Select`, this is confirmed with `COT_deactcon` (9) and the `Select` is deleted.
- ▶ If an `Execute` or `Cancel` is received without a `Select` beforehand, it is responded to with `PN` (i.e. negative).

Pulse generation - 'Qualifier of Command'

The IEC870 slave supports pulse generation for commands received. The Qualifier of Command (QoC) is supported for all commands envisaged in the IEC60870 standard (T45, T46, T47 and T58, T59, T60).

For example, a pulse command received - an ASDU of type T45 - with QoC has the following effect on a zenon variable:

- ▶ **QoC 0:** Undefined in accordance with IEC standard. Behaves in slave as in QoC 3 persistent output.
- ▶ **QoC 1:** Short pulse. The zenon variable assumes the value 'ON' and changes, after the time defined in the Process Gateway settings (on page 161) to 'OFF'.
- ▶ **QoC 2:** Long pulse. The zenon variable assumes the value 'ON' and changes, after the time defined in the Process Gateway settings (on page 161) to 'OFF'.
- ▶ **QoC 3:** Persistent output. The zenon variable assumes the value 'ON' (or 'OFF') and retains the value. This must be reset to receive further commands.

As soon as communication is ended, the pulse for all other active pulse variables will be stopped (i.e. an OFF will be written). If a pulse is triggered for a running pulse generation, the active pulse generation will be stopped. The variable contains the value 'OFF'.

Note: For SCS (single command state) the value for 'OFF' is always 0. The following is applicable for DCS (double command state): An 'OFF' has the value 00b at protocol level; then either the value 2 or 0 in zenon, depending on the settings for **DPI/DCS mapping** (on page 190).

Note: In sectors with `Select routing`, pulse generation cannot be carried out because 'OFF' after `Select` is in conflict with `Execute`.

10.3.10 Read - T102 - request

The IEC870 slave automatically responds to the Read command - `C_RD_NA_1` - requests from the master. No additional project configuration steps are necessary for this.

A maximum of 2048 read requests can be active at the same time. Once this limit has been reached, further read requests are rejected.

The response to the read requests does not need to be configured and automatically runs in the background. For this, the following applies:

- ▶ For each IO with the requested `IOA`, a read is executed in zenon and a response is sent to the master with the value provided by Runtime.
- ▶ Several requests for one IO can be active at the same time.
- ▶ Read requests will also be processed even if the general request is not finished yet.

10.3.11 Time synchronization - T103 - request

In order for time synchronization to be carried out - on receipt of `C_CS_NA_1`, there must be in Device, at any desired Sektor a type `T103` IO connected to a process variable. The first IO that uses type `T103` is used as an enable.

This variable must have a value of $\neq 0$ and must not have any of the following status bits set:

- ▶ INVALID (Bit 18)
- ▶ OFF (Bit 20)
- ▶ ALT_VAL (Bit 27)
- ▶ OV_870 (Bit 47)
- ▶ BL_870 (Bit 44)

If the `Enable` variable has not been configured or the value for this variable is missing, the connection is disconnected if a `C_CS_NA_1` is received, because it indicates that the system has not been fully configured.

Non-execution of the command due to a missing enable (value 0 or status) or rights in the operating system does not lead to the connection being terminated, the command is only confirmed negatively - `COT_actcon + PN`.

Note: to convert the computer's clock, the Process Gateway application needs administrator rights in Windows.

10.3.12 Test - T104 and T107 - requests

The IEC870 slave responds to the master automatically with the two test commands `C_TS_NA_1` (ASDU<104>) and `C_TS_TA_1` (ASDU<107>). No additional project configuration steps are necessary for this.



Information

You can find further information on supported commands in the Interoperability (on page 232) chapter.

10.3.13 File transfer

The **Process Gateway IEC870 Slave** supports the transfer of files in monitoring and controlling direction. To do this, configure, in **Device**, the folder property **Directory for File Transfer** and **Directory for File Transfer in Control Direction**.

The procedures of file transfer with the 870 master are automatically processed in the background. The ASDUs used are sent as class 2. This ensures that value changes are also transferred to the master during an ongoing file transfer.

The following is applicable for the ASDUs of the file transfer:

- ▶ The values 1-4 are supported in the AFQ UI4
- ▶ The values 0-6 are supported in the SCQ UI4;
0 and 1 are both 'select file'

If a new transfer is started while another one is still active, the currently active transfer will be interrupted and the new transfer will be started.

Encrypted communication (on page 222) of the **IEC870 Slave Process Gateway** is configured in the `zenProcGateway.INI` file.

Note: The current status of the file transfer can be evaluated using variables that are linked to the internal type T00 (on page 210).

Storage and naming

Use the following scheme for storage and naming:

<Configured folder>\<COA>\<IOA>.<NOF>

Only 1 = 'Name of file' is supported as NOF (transparent file).
The maximum size of files for transfer is 16.711.680 Bytes.

EXAMPLE

Folder in the module configuration (on page 161): `C:\TEMP\IEC870`

File 1100.1: For IO of sector **COA=151** and **IOA 1100**

Save location: `C:\TEMP\IEC870\151\1100.1`

Spontaneous directory transfer

Spontaneous transfer of files in the folder for file transfer can be triggered by the process control system. To do this:

- create an IO of type F_DR_NA_1 (T126) with IOA '0'.

Value	Meaning
Change from 1 to 2	Starting transfer
3	Transfer active
4	Transfer finished
5	Transfer failure

Nested activation is prevented.

TLS encrypted communication

Encrypted communication for the **IEC870 Slave Process Gateway** is configured in the `zenProcGateway.INI` file.

Parameter	Description
TLS_ACTIVE=	<p>Type of secure communication</p> <ul style="list-style-type: none"> ▶ 0: deactivated No encrypted communication ▶ 1: activated Communication is in encrypted form on the basis of the following parameters <p>Default: 0</p>
TLS_CERTIFICATE_STORE_PATH=	<p>Absolute save location of the certificates. This basic directory must contain the two subdirectories CA and PRIVATE .</p> <ul style="list-style-type: none"> ▶ CA for trusted certificates <p>and</p> <ul style="list-style-type: none"> ▶ PRIVATE for your own certificates
TLS_CERTIFICATE_FILE=	<p>Name of the TLS certificate.</p> <p>The default save location is configured with the entry for the [save location of the certificates] (TLS_CERTIFICATE_STORE_PATH=) . The TLS certificate must be present in the PRIVATE folder.</p>
TLS_CIPHER_LIST=	<p>List of the supported encryption processes. The list contains an abbreviation in openssl format.</p>
TLS_PEER_CERTIFICATE_SUBJECT=	<p>Client certificate identification used.</p> <p>The client certificate must contain the SUBJECT configured here. The connection is disconnected if they do not correspond.</p>
TLS_RENEGOTIATION_TIMEOUT=	<p>Maximum duration of an encrypted connection before it is renewed.</p> <p>The encryption is reinitialized after the configured time has expired. In doing so, certificates and the key are replaced for new ones. Time indication in seconds.</p> <p>Default: 86400</p>
TLS_RENEGOTIATION_MAX_BYTES=	<p>Amount of data for which an encrypted connection is used before it is renewed.</p> <p>The encryption is reinitialized after the configured amount of data has been reinitialized. In doing so, certificates and the key are replaced for new ones. Indication in bytes.</p> <p>Default: 1048576</p>
TLS_RESUMPTION_TIMEOUT=	<p>Maximum duration of a key used before it is replaced.</p>

	<p>Only the key is renewed. The certificate is not updated in the process. The certificate is renewed after expiry of the configured time. Time indication in seconds.</p> <p>Default: 43200</p>
TLS_RESUMPTION_MAX_BYTES=	<p>Amount of data for which a key is used before it is renewed.</p> <p>The encryption is renewed after the configured amount of data has been transferred. The certificate is not updated in the process. Indication in bytes.</p> <p>Default: 10485760</p>
TLS_CRL_CHECK_INTERVAL=21600	<p>Time interval for the check to see whether the certificate currently being used is included in the REVOCATION LIST.</p> <p>The encrypted connection is no longer secure if the certificate is included in the REVOCATION LIST. The connection is terminated. Time indication in seconds.</p> <p>Default: 21600</p>

10.4 Broadcasts

Global address 0xFF(FF) in the Common Address of ASDU (COA) is supported for the following Typ IDs:

- ▶ C_IC_NA_1 - General query - ASDU<100>
- ▶ C_CS_NA_1 - Time synchronization - ASDU<103>

No additional project configuration steps are necessary for this.



Attention

The Broadcast Address 0xFF(FF) - "all outstations" - in Data Link Layer in 60870-4-101 communication is not supported.

10.5 Error analysis

Should there be communication problems, this chapter will assist you in finding out the error.

10.5.1 Analysis tool

Process Gateway behaves like a zenon driver and writes messages to a common log file. To display them correctly and clearly, use the Diagnosis Viewer (main.chm::/12464.htm) program that was also installed with zenon. You can find it under Start/All programs/zenon/Tools 8.00 -> Diagviewer. In general:

zenon driver log all errors in the LOG files. LOG files are text files with a special structure. The default folder for the LOG files is subfolder **LOG** in the folder **ProgramData**. For example:

%ProgramData%\COPA-DATA\LOG.

Attention: With the default settings, a driver only logs error information. With the Diagnosis Viewer you can enhance the diagnosis level for most of the drivers to "Debug" and "Deep Debug". With this the driver also logs all other important tasks and events.

In the Diagnosis Viewer you can also:

- ▶ Follow newly-created entries in real time
- ▶ customize the logging settings
- ▶ change the folder in which the LOG files are saved

Note:

1. The Diagnosis Viewer displays all entries in UTC (coordinated world time) and not in local time.
2. The Diagnosis Viewer does not display all columns of a LOG file per default. To display more columns activate property **Add all columns with entry** in the context menu of the column header.
3. If you only use **Error-Logging**, the problem description is in the column **Error text**. For other diagnosis level the description is in the column **General text**.
4. For communication problems many drivers also log error numbers which the PLC assigns to them. They are displayed in **Error text** or **Error code** or **Driver error parameter** (1 and 2). Hints on the meaning of error codes can be found in the driver documentation and the protocol/PLC description.
5. At the end of your test set back the diagnosis level from **Debug** or **Deep Debug**. At **Debug** and **Deep Debug** there are a great deal of data for logging which are saved to the hard drive and which can influence your system performance. They are still logged even after you close the Diagnosis Viewer.



Attention

In Windows CE errors are not logged per default due to performance reasons.

Note: Process Gateway is not available under Windows CE.



Information

You can find further information on the Diagnosis Viewer in the *Diagnosis Viewer* (main.chm::/12464.htm) chapter.

Logging

The driver supports a comprehensive logging function for:

- ▶ Client name
- ▶ zenProcGateway_IEC870Slave
- ▶ Modules

CLIENT NAME

The way the name of the client is displayed in the Diagnosis Viewer depends on the configuration. That is why the name changes if you change the TCP port, the COM port or the protocol. In this case, the logging settings may also change.

If the Process Gateway is started several times with the same settings, the same client name will be used. The error messages of the different instances can be distinguished by the process ID.



Example

zenProcGateway_IEC870Slave

- ▶ 104 protocol with standard port 2404
- ▶ 101 if COM1 selected.
- ▶ If the configuration has not been loaded yet, error messages during startup will always be visible under this client.

zenProcGateway_IEC870Slave:COM4

- ▶ 101 with COM4.
The name contains the used COM port.

zenProcGateway_IEC870Slave:Port:12345

- ▶ 104 with port 12345.
The name contains the port for the Master connection.

MODULES

Error messages are shown for all modules:

- ▶ IEC870 Slave
- ▶ OSI2
- ▶ OSI7
- ▶ SCADA Runtime (ScadaRT)

IEC870 SLAVE

Concerns the connection to the Process Gateway and its configuration.

Information about:

- ▶ Errors during engineering

OSI2

Concerns the protocol level. This where framing with its according functions is implemented.

Information about:

- ▶ Received data as byte dump
- ▶ Sent data as byte dump
- ▶ Function codes for 101
- ▶ Errors on this level, e.g. in the framing
- ▶ APCI for 104 (870-5-104 5)

Levels

- ▶ Msg: Frame interpreting
- ▶ Debug:
Received and sent data
Frame analysis.
New Master connection.
- ▶ Deep Debug: Every request about existence of characters. Produces a large number of logging entries!

OSI7

Concerns the application layer.

This is where commands are interpreted and executed.

Information about:

- ▶ Logging 'process and system information'
- ▶ Errors in the ASDUs

Levels:

- ▶ Warning: Recursive interrogation
- ▶ Msg: Process information, System information
- ▶ Debug: Confirmed I-frames
- ▶ Deep Debug: Interrogation runs. Produces a large number of logging entries!

SCADA RUNTIME (SCADART)

Messages for connection to the Runtime:

Information about:

- ▶ advised and unadvised variables
- ▶ updated variables
- ▶ Deactivating the project

Levels

- ▶ Debug:
changed values
project inactive

LOG entries

The IEC870 slave uses the following parameters in entries in the LOG file:

Parameter	Description
deviceIp:	IP address from the device
devicename:	Name of device
COA:	Care of Address -> according to standard
SecName :	Sector description
TypeyId:	Command number
TypeIdAsTxt:	Command as text
IOA:	According to standard
AdviseID:	ID via which value changes are requested and assigned

Value:	Value transferred
TimeStamp:	Time of value change. From the driver or command
QUALDESC:	According to standard
SBEValue:	COT, select and pn mapped to unique enums: 0: Not select relevant 1: actcon+pn+select 2: actcon+select 3: deactcon+pn+select, 4: deactcon+select 5: actterm+pn+select, 6: actterm+select,
Confirmed:	1 if select already confirmed
COT:	Cause of transmission in accordance with standard
CotTxt:	COT as text
Qualifier:	According to standard and command
Originator:	According to standard
QU:	Qualifier of command QU 7.2.6.26
DCS:	7.2.6.16 Double command (IEV 371-03-03)
ControlCommand:	Depending on control command

EXAMPLE FOR 45:

```

<TypeId> <TypeIdAsTxt> Device:(<deviceip>)<devicename> Sector:(<IOA>)< SecName >
ioa:<IOA> cot:(<COT>)<CotTxt> ori:<Originator> sel:<QOC S/E> qu:<QU> dcs:<DCS>
ValueChange cmd(<TypeeyId>)<<TypeId>> <TypeIdAsTxt> ioa:<IOA> id:<AdviseID> value:<Wert>
time:<TimeStamp> state:< QUALDESC > sir:<SBEValue>

```

LOG entry	Module/Level	Description
SBE IED processing. Device: (<deviceIp>) <devicename> Sector: (<COA>) <SecName> <ValueChange> SBE command: advise: <AdviseID> conf: <confirmed> <ControlCommand>	OSI7/Debug	Each value change that is processed for SBE creates this log message
Wrn:1014 SBE activation already confirmed! Device: (<DeviceIp>) <devicename> Sector: (<COA>) <SecName> <ValueChange>	OSI7/Warning	Renewed conformation for Select received and ignored
Error:16 Advise for SBE data point fail. SBE command: advise: <AdviseId> conf: <confirmed> <ControlCommand>	Scada/error	Variable could not be requested.
Error:17 SOE value lost, queue full! Device: (<DeviceIp>) <devicename> Sector: (<COA>) <SecName> <ValueChange>	Scada/Error	Value change for SOE lost because the source is full.

10.5.2 Error codes

The following error codes can be displayed in the Diagnosis Viewer:

Error code	Meaning
570	Value change for unknown cyclical IO.
571	Transfer buffer too small to transfer a cyclical IO.
576	Loss of connection. (Master does not respond.)

ERROR MESSAGES FOR FILE TRANSFER IN REVERSE DIRECTION

Error number	Error level	Description
20	Error message	Creation of the temporary file for the file transfer in reverse direction was unsuccessful.
1017	Warning	File transfer in reverse direction is already active, the ongoing one will be ended and a new one started.
21	Error message	Segment data write error occurred
22	Error message	Length of the data exceeds the expected file length
23	Error message	Checksum of the transferred segments incorrect for the sector.
24	Error message	Checksum for file incorrect.
25	Error message	Segment data transferred for sector exceeds the data length defined at the start
26	Error message	Sector data transfer is reported as finished but the amount of segment data transferred does not correspond to that stated at the start.
27	Error message	File data is reported as completed but the given file length does not correspond.
581	Error message	Unexpected IOA with SG
582	Error message	Unexpected NOF with SG
583	Error message	Unexpected NOS with SG
584	Error message	Unexpected NOS with LS
585	Error message	Unexpected NOS with LS
586	Error message	Unexpected NOS with LS
587	Error message	FR with non-supported NOF. We only support 1 == transparent

10.5.3 Check list

870-104

870-101

1. Is the COM port in use by another application or are the settings incorrect?

General

- 1.
2. Is the device (PLC) that you are trying to communicate with connected to the power supply?
3. Is the cable between PLC and PC/IPC connected correctly?
4. Have you analyzed the error file (which errors did occur)?
5. For further error analysis, please send a project backup, the INI and XML files of the IEC870 slave and the LOG file of the **Diagnosis Viewer** to the support team responsible for you.

10.6 Interoperability

This companion standard presents sets of parameters and alternatives from which subsets must be selected to implement particular telecontrol systems. Certain parameter values, such as the choice of 'structured' or 'unstructured' fields of the information object address of ASDUs represent mutually exclusive alternatives. This means that only one value of the defined parameters is admitted per system. Other parameters, such as the listed set of different process information in command and in monitor direction allow the specification of the complete set or subsets, as appropriate for given applications. This clause summarizes the parameters of the previous clauses to facilitate a suitable selection for a specific application. If a system is composed of equipment stemming from different manufacturers, it is necessary that all partners agree on the selected parameters.

The interoperability list is defined as in IEC 60870-5-101 and extended with parameters used in this standard. The text descriptions of parameters which are not applicable to this companion standard are strike-through (corresponding check box is marked black).

NOTE In addition, the full specification of a system may require individual selection of certain parameters for certain parts of the system, such as the individual selection of scaling factors for individually addressable measured values.

The selected parameters should be marked in the white boxes as follows:

- [] Function or ASDU is not used
- [X] Function or ASDU is used as standardized (default)
- [R] Function or ASDU is used in reverse mode
- [B] Function or ASDU is used in standard and reverse mode

The possible selection (blank, X, R, or B) is specified for each specific clause or parameter.

A black check box indicates that the option cannot be selected in this companion standard.

1. SYSTEM OR DEVICE

(system-specific parameter, indicate definition of a system or a device by marking one of the following with 'x')

- ☐ System definition
- ☐ Controlling station definition (Master)
- ☒ Controlled station definition (Slave)

2. NETWORK CONFIGURATION: 101 ONLY

(network-specific parameter, all configurations that are used are to be marked 'x')

<input checked="" type="checkbox"/> Point-to-point	<input checked="" type="checkbox"/> Multipoint
<input checked="" type="checkbox"/> Multipoint to point	<input type="checkbox"/> Multipoint-star

3. PHYSICAL LAYER: 101 ONLY

(network-specific parameter, all interfaces and data rates that are used are to be marked 'x')

TRANSMISSION SPEED (CONTROL DIRECTION)

Unbalanced interchange Circuit V.24/V.28 Standard	Unbalanced interchange Circuit V.24/V.28 Recommended if >1 200 bit/s	Balanced interchange Circuit X.24/X.27
<input type="checkbox"/> 100 bit/s	<input checked="" type="checkbox"/> 2400 bit/s	<input checked="" type="checkbox"/> 2400 bit/s
<input type="checkbox"/> 200 bit/s	<input checked="" type="checkbox"/> 4800 bit/s	<input checked="" type="checkbox"/> 4800 bit/s
<input checked="" type="checkbox"/> 300 bit/s	<input checked="" type="checkbox"/> 9600 bit/s	<input checked="" type="checkbox"/> 9600 bit/s
<input checked="" type="checkbox"/> 600 bit/s	<input checked="" type="checkbox"/> 19200 bit/s	<input checked="" type="checkbox"/> 19200 bit/s
<input checked="" type="checkbox"/> 1200 bit/s	<input checked="" type="checkbox"/> 38400 bit/s	<input checked="" type="checkbox"/> 38400 bit/s
	<input checked="" type="checkbox"/> 56000 bit/s	<input checked="" type="checkbox"/> 56000 bit/s
	<input checked="" type="checkbox"/> 57600 bit/s	<input checked="" type="checkbox"/> 57600 bit/s
	<input checked="" type="checkbox"/> 115200 bit/s	<input checked="" type="checkbox"/> 115200 bit/s
	<input checked="" type="checkbox"/> 128000 bit/s	<input checked="" type="checkbox"/> 128000 bit/s
	<input checked="" type="checkbox"/> 256000 bit/s	<input checked="" type="checkbox"/> 256000 bit/s

TRANSMISSION SPEED (MONITOR DIRECTION)

Unbalanced interchange Circuit V.24/V.28 Standard	Unbalanced interchange Circuit V.24/V.28 Recommended if >1 200 bit/s	Balanced interchange Circuit X.24/X.27
[] 100 bit/s	[X] 2400 bit/s	[X] 2400 bit/s
[] 200 bit/s	[X] 4800 bit/s	[X] 4800 bit/s
[X] 300 bit/s	[X] 9600 bit/s	[X] 9600 bit/s
[X] 600 bit/s	[X] 19200 bit/s	[X] 19200 bit/s
[X] 1200 bit/s	[X] 38400 bit/s	[X] 38400 bit/s
	[X] 56000 bit/s	[X] 56000 bit/s
	[X] 57600 bit/s	[X] 57600 bit/s
	[X] 115200 bit/s	[X] 115200 bit/s
	[X] 128000 bit/s	[X] 128000 bit/s
	[X] 256000 bit/s	[X] 256000 bit/s

4. LINK LAYER: 101 ONLY

(network-specific parameter, all options that are used are to be marked 'x'. Specify the maximum frame length. If a non-standard assignment of class 2 messages is implemented for unbalanced transmission, indicate the Type ID and COT of all messages assigned to class 2.)

Frame format FT 1.2, single character 1 and the fixed time out interval are used exclusively in this companion standard.

Link transmission	Frame length [octets]	Address field of the link
[X] Balanced transmission	[255] Maximum length L (both directions)*	[O] not present (balanced transmission only) optional
[X] Unbalanced transmission		[X] One octet
		[X] Two octets
		[X] Structured
		[X] Unstructured

*may be reduced by the system

Link Address 0xFF (FF) - broadcast to all outstations - is not supported.

The structure of Link Address is not interpreted.

When using an unbalanced link layer, the following ASDU types are returned in class 2 messages (low priority) with the indicated causes of transmission:

[] The standard assignment of ASDUs to class 2 messages is used as follows:

Type identification	Cause of transmission
9, 11, 13, 21	<1>

[x] A special assignment of ASDUs to class 2 messages is used as follows:

Type identification	Cause of transmission
120 - 126	All (as specified in the standard)

Note: (In response to a class 2 poll, a controlled station may respond with class 1 data when there is no class 2 data available).

5. APPLICATION LAYER

TRANSMISSION MODE FOR APPLICATION DATA

Mode 1 (Least significant octet first), as defined in 4.10 of IEC 60870-5-4, is used exclusively in this companion standard.

COMMON ADDRESS OF ASDU

(system-specific parameter, all configurations that are used are to be marked ' x ')

ASDU address	
101 only	[x] Two octets
[x] One octet	

The ASDU address 0xFF(FF) - global address - is supported only for C_IC_NA_1 and C_CS_NA_1

INFORMATION OBJECT ADDRESS

(system-specific parameter, all configurations that are used are to be marked ' x ')

Object address	
101 only	[x] Structured
[x] One octet	
101 only	[x] Unstructured

[X] Two octets	
[X] Three octets	

CAUSE OF TRANSMISSION

(system-specific parameter, all configurations that are used are to be marked ' X ')

Cause of transmission	
101 only	[X] Two octets (with originator address) Originator address is set to zero if not used.
[X] One octet	

LENGTH OF APDU: 104 ONLY

(system-specific parameter, specify the maximum length of the APDU per system)

The maximum length of APDU for both directions is 253. The maximum length may be reduced by the system.

[253] Maximum length of APDU per system

SELECTION OF STANDARD ASDUS

PROCESS INFORMATION IN MONITOR DIRECTION

(station-specific parameter, mark each Type ID ' X ' if it is only used in the standard direction, ' R ' if only used in the reverse direction, and ' B ' if used in both directions).

Mark	Parameter	Type
[B]	<1>: = Single-point information	M_SP_NA_1
[B]	<2>: = Single-point information with time TAG	M_SP_TA_1
[B]	<3>: = Double-point information	M_DP_NA_1
[B]	<4>: = Double-point information with time TAG	M_DP_TA_1
[B]	<5>: = Step position information	M_ST_NA_1
[B]	<6>: = Step position information with time TAG	M_ST_TA_1
[B]	<7>: = Bitstring of 32 bit	M_BO_NA_1
[B]	<8>: = Bitstring of 32 bit with time TAG	M_BO_TA_1
[B]	<9>: = Measured value, normalized value	M_ME_NA_1
[B]	<10>: = Measured value, normalized value with time TAG	M_ME_TA_1
[B]	<11>: = Measured value, scaled value	M_ME_NB_1
[B]	<12>: = Measured value, scaled value with time TAG	M_ME_TB_1
[B]	<13>: = Measured value, short floating point value	M_ME_NC_1
[B]	<14>: = Measured value, short floating point value with time TAG	M_ME_TC_1
[B]	<15>: = Integrated totals	M_IT_NA_1
[B]	<16>: = Integrated totals with time TAG	M_IT_TA_1
[]	<17>: = Event of protection equipment with time TAG	M_EP_TA_1
[]	<18>: = Packed start events of protection equipment with time TAG	M_EP_TB_1
[]	<19>: = Packed output circuit information of protection equipment with time TAG	M_EP_TC_1
[]	<20>: = Packed single-point information with status change detection	M_SP_NA_1
[]	<21>: = Measured value, normalized value without quality descriptor	M_ME_ND_1
[B]	<30>: = Single-point information with time TAG CP56Time2a	M_SP_TB_1
[B]	<31>: = Double-point information with time TAG CP56Time2a	M_DP_TB_1
[B]	<32>: = Step position information with time TAG CP56Time2a	M_ST_TB_1
[B]	<33>: = Bitstring of 32 bit with time TAG CP56Time2a	M_BO_TB_1
[B]	<34>: = Measured value, normalized value with time TAG CP56Time2a	M_ME_TD_1
[B]	<35>: = Measured value, scaled value with time TAG CP56Time2a	M_ME_TE_1
[B]	<36>: = Measured value, short floating point value with time TAG CP56Time2a	M_ME_TF_1
[B]	<37>: = Integrated totals with time TAG CP56Time2a	M_IT_TB_1
[]	<38>: = Event of protection equipment with time TAG CP56Time2a	M_EP_TD_1

[]	<39>:= Packed start events of protection equipment with time TAG CP56Time2a	M_EP_TE_1
[]	<40>:= Packed output circuit information of protection equipment with time TAG CP56Time2a	M_EP_TF_1

Either the ASDUs of the set <2>, <4>, <6>, <8>, <10>, <12>, <14>, <16>, <17>, <18>, <19> or of the set <30> – <40> are used.

PROCESS INFORMATION IN CONTROL DIRECTION

(station-specific parameter, mark each Type ID 'X' if it is only used in the standard direction, 'R' if only used in the reverse direction, and 'B' if used in both directions).

Mark	Parameter	Type
[X]	<45>:= Single command	C_SC_NA_1
[X]	<46>:= Double command	C_DC_NA_1
[X]	<47>:= Regulating step command	C_RC_NA_1
[X]	<48>:= Set point command, normalized value	C_SE_NA_1
[X]	<49>:= Set point command, scaled value	C_SE_NB_1
[X]	<50>:= Set point command, short floating point value	C_SE_NC_1
[X]	<51>:= Bitstring of 32 bit	C_BO_NA_1
[X]	<58>:= Single command with time TAG CP56Time2a	C_SC_TA_1
[X]	<59>:= Double command with time TAG CP56Time2a	C_DC_TA_1
[X]	<60>:= Regulating step command with time TAG CP56Time2a	C_RC_TA_1
[X]	<61>:= Set point command, normalized value with time TAG CP56Time2a	C_SE_TA_1
[X]	<62>:= Set point command, scaled value with time TAG CP56Time2a	C_SE_TB_1
[X]	<63>:= Set point command, short floating point value with time TAG CP56Time2a	C_SE_TC_1
[X]	<64>:= Bitstring of 32 bit with time TAG CP56Time2a	C_BO_TA_1

Either the ASDUs of the set <45> – <51> or of the set <58> – <64> are used.

SYSTEM INFORMATION IN MONITOR DIRECTION

(station-specific parameter, mark 'X' if used)

Mark	Parameter	Type
[X]	<70> := End of initialization	M_EI_NA_1

SYSTEM INFORMATION IN CONTROL DIRECTION

(station-specific parameter, mark each Type ID 'X' if it is only used in the standard direction, 'R' if only used in the reverse direction, and 'B' if used in both directions).

Mark	Parameter	Type
[X]	<100>: = Interrogation command*	C_IC_NA_1
[X]	<101>: = Counter interrogation command	C_CI_NA_1
[X]	<102>: = Read command	C_RD_NA_1
[X]	<103>: = Clock synchronization command (option see 7.6)*	C_CS_NA_1
[X]	<104>: = Test command	C_TS_NA_1
[X]	<105>: = Reset process command	C_RP_NA_1
[]	<106>: = Delay acquisition command	C_CD_NA_1
[X]	<107>: = Test command with time TAG CP56Time2a	C_TS_TA_1

*also with global address - with Common Address of ASDU = 0xFF(FF)

PARAMETER IN CONTROL DIRECTION

(station-specific parameter, mark each Type ID 'X' if it is only used in the standard direction, 'R' if only used in the reverse direction, and 'B' if used in both directions).

Mark	Parameter	Type
[]	<110>: = Parameter of measured value, normalized value	P_ME_NA_1
[]	<111>: = Parameter of measured value, scaled value	P_ME_NB_1
[]	<112>: = Parameter of measured value, short floating point value	P_ME_NC_1
[]	<113>: = Parameter activation	P_AC_NA_1

FILE TRANSFER

(station-specific parameter, mark each Type ID 'X' if it is only used in the standard direction, 'R' if only used in the reverse direction, and 'B' if used in both directions).

Mark	Parameter	Type
[B]	<120>: = File ready	F_FR_NA_1
[B]	<121>: = Section ready	F_SR_NA_1
[B]	<122>: = Call directory, select file, call file, call section	F_SC_NA_1
[B]	<123>: = Last section, last segment	F_LS_NA_1
[B]	<124>: = Ack file, ack section	F_AF_NA_1
[B]	<125>: = Segment	F_SG_NA_1
[X]	<126>: = Directory {blank or X, only available in monitor (standard) direction}	F_DR_TA_1

TYPE IDENTIFIER AND CAUSE OF TRANSMISSION ASSIGNMENTS

(station-specific parameters)

Shaded boxes: option not required.

Blank: functions or ASDU not used.

Mark Type Identification/Cause of transmission combinations:

' X ' if only used in the standard direction;

' R ' if only used in the reverse direction;

' B ' if used in both directions.

Type identification		Cause of transmission																		
		1	2	3	4	5	6	7	8	9	10	11	12	13	20 to 36	37 to 41	44	45	46	47
<1>	M_SP_NA_1		B	B		X									X					
<2>	M_SP_TA_1			B		X														
<3>	M_DP_NA_1		B	B		X									X					
<4>	M_DP_TA_1			B		X														
<5>	M_ST_NA_1		B	B		X									X					
<6>	M_ST_TA_1			B		X														
<7>	M_BO_NA_1		B	B		X									X					
<8>	M_BO_TA_1			B		X														
<9>	M_ME_NA_1	B	B	B		X									X					
<10>	M_ME_TA_1			B		X														
<11>	M_ME_NB_1	B	B	B		X									X					
<12>	M_ME_TB_1			B		X														
<13>	M_ME_NC_1	B	B	B		X									X					
<14>	M_ME_TC_1			B		X														
<15>	M_IT_NA_1			B		X									X ¹	X				
<16>	M_IT_TA_1			B		X										X				
<17>	M_EP_TA_1																			
<18>	M_EP_TB_1																			
<19>	M_EP_TC_1																			
<20>	M_PS_NA_1																			
<21>	M_ME_ND_1																			
<30>	M_SP_TB_1			B		X														
<31>	M_DP_TB_1			B		X														
<32>	M_ST_TB_1			B		X														
<33>	M_BO_TB_1			B		X														
<34>	M_ME_TD_1			B		X														
<35>	M_ME_TE_1			B		X														
<36>	M_ME_TF_1			B		X														
<37>	M_IT_TB_1			B		X										X				

Type identification		Cause of transmission																		
		1	2	3	4	5	6	7	8	9	10	11	12	13	20 to 36	37 to 41	44	45	46	47
<38>	M_EP_TD_1																			
<39>	M_EP_TE_1																			
<40>	M_EP_TF_1																			
<45>	C_SC_NA_1						X	X	X²	X²	X								X	X
<46>	C_DC_NA_1						X	X	X²	X²	X								X	X
<47>	C_RC_NA_1						X	X	X²	X²	X								X	X
<48>	C_SE_NA_1						X	X	X²	X²	X								X	X
<49>	C_SE_NB_1						X	X	X²	X²	X								X	X
<50>	C_SE_NC_1						X	X	X²	X²	X								X	X
<51>	C_BO_NA_1						X	X	X²	X²	X								X	X
<58>	C_SC_TA_1						X	X	X²	X²	X								X	X
<59>	C_DC_TA_1						X	X	X²	X²	X								X	X
<60>	C_RC_TA_1						X	X	X²	X²	X								X	X
<61>	C_SE_TA_1						X	X	X²	X²	X								X	X
<62>	C_SE_TB_1						X	X	X²	X²	X								X	X
<63>	C_SE_TC_1						X	X	X²	X²	X								X	X
<64>	C_BO_TA_1						X	X	X²	X²	X								X	X
<70>	M_EI_NA_1*				X															
<100>	C_IC_NA_1						X	X	X	X	X							X	X	
<101>	C_CI_NA_1						X	X			X							X	X	
<102>	C_RD_NA_1					X												X	X	X
<103>	C_CS_NA_1						X	X										X	X	
<104>	C_TS_NA_1						X	X												
<105>	C_RP_NA_1						X	X										X	X	
<106>	C_CD_NA_1																			
<107>	C_TS_TA_1						X	X												
<110>	P_ME_NA_1																			
<111>	P_ME_NB_1																			
<112>	P_ME_NC_1																			

Type identification		Cause of transmission																			
		1	2	3	4	5	6	7	8	9	10	11	12	13	20 to 36	37 to 41	44	45	46	47	
<113>	P_AC_NA_1																				
<120>	F_FR_NA_1													X					X		
<121>	F_SR_NA_1													X					X		
<122>	F_SC_NA_1					X								X				X	X		
<123>	F_LS_NA_1													X					X		
<124>	F_AF_NA_1													X				X	X		
<125>	F_SG_NA_1													X					X		
<126>	F_DR_TA_1*			X		X															

* Blank or X only

¹ Optional.

² Slave confirms deactivations of write command (TI <45> ... <64>) but by deactivated '**Select routing**' the corresponding activation is already transferred to execution; COT_actcon and COT_actterm were already sent.

COT	Cause of Transmission	
<0>	not used	
<1>	periodic, cyclic	per/cyc
<2>	background scan	back
<3>	spontaneous	spont
<4>	initialized	init
<5>	request or requested	req
<6>	activation	act
<7>	activation confirmation	actcon
<8>	deactivation	deact
<9>	deactivation confirmation	deactcon
<10>	activation termination	actterm
<11>	return information caused by a remote command	retrem
<12>	return information caused by a local command	retloc
<13>	file transfer	file
<14...19>	reserved	
<20>	interrogated by station interrogation	inrogen
<21...36>	interrogated by interrogation of the group 1..16	inro1..16
<37>	requested by general counter request	reqcogen
<38...41>	requested by counter interrogation of the group 1 ... 4	reqco1..4
<42, 43>	reserved	
<44>	unknown type identification	
<45>	unknown cause of transmission	
<46>	unknown common address of ASDU	
<47>	unknown information object address	
<48, 63>	for special use (private range)	

6. BASIC APPLICATION FUNCTIONS

STATION INITIALIZATION

(station-specific parameter, mark 'X' if function is used)

[] Remote initialization

CYCLIC DATA TRANSMISSION

(station-specific parameter, mark 'X' if function is only used in the standard direction, 'R' if only used in the reverse direction, and 'B' if used in both directions)

[X] Cyclic data transmission

READ PROCEDURE

(station-specific parameter, mark 'X' if function is only used in the standard direction, 'R' if only used in the reverse direction, and 'B' if used in both directions)

[X] Read procedure

SPONTANEOUS TRANSMISSION

(station-specific parameter, mark 'X' if function is only used in the standard direction, 'R' if only used in the reverse direction, and 'B' if used in both directions)

[B] Spontaneous transmission

DOUBLE TRANSMISSION OF INFORMATION OBJECTS WITH CAUSE OF TRANSMISSION SPONTANEOUS

(station-specific parameter, mark each information type 'X' where both a Type ID without time and corresponding Type ID with time are issued in response to a single spontaneous change of a monitored object)

The following type identifications may be transmitted in succession caused by a single status change of an information object. The particular information object addresses for which double transmission is enabled are defined in a project-specific list.

- [] Single-point information M_SP_NA_1, M_SP_TA_1, M_SP_TB_1 and M_PS_NA_1
- [] Double-point information M_DP_NA_1, M_DP_TA_1 and M_DP_TB_1
- [] Step position information M_ST_NA_1, M_ST_TA_1 and M_ST_TB_1
- [] Bitstring of 32 bit M_BO_NA_1, M_BO_TA_1 and M_BO_TB_1 (if defined for a specific project)
- [] Measured value, normalized value M_ME_NA_1, M_ME_TA_1, M_ME_ND_1 and M_ME_TD_1
- [] Measured value, scaled value M_ME_NB_1, M_ME_TB_1 and M_ME_TE_1
- [] Measured value, short floating point number M_ME_NC_1, M_ME_TC_1 and M_ME_TF_1

STATION INTERROGATION

(station-specific parameter, mark 'X' if function is only used in the standard direction, 'R' if only used in the reverse direction, and 'B' if used in both directions).

--	--	--	--

[X] global		
[] group 1	[] group 7	[] group 13
[] group 2	[] group 8	[] group 14
[] group 3	[] group 9	[] group 15
[] group 4	[] group 10	[] group 16
[] group 5	[] group 11	Information object addresses assigned to each group must be shown in a separate table.
[] group 6	[] group 12	

CLOCK SYNCHRONIZATION

(station-specific parameter, mark 'X' if function is only used in the standard direction, 'R' if only used in the reverse direction, and 'B' if used in both directions).

[X] Clock synchronization

[] Day of week used

[] RES1, GEN (time tag substituted/not substituted) used

[B] SU-bit (summertime) used

optional, see 7.6

COMMAND TRANSMISSION

(station-specific parameter, mark 'X' if function is only used in the standard direction, 'R' if only used in the reverse direction, and 'B' if used in both directions).

[X] Direct command transmission

[X] Direct set point command transmission

[X] Select and execute command

[X] Select and execute set point command

[X] C_SE ACTTERM used

[X] No additional definition

[X] Short-pulse duration (duration determined by a system parameter in the outstation)

[X] Long-pulse duration (duration determined by a system parameter in the outstation)

[X] Persistent output

[setting] Supervision of maximum delay in command direction of commands and set point commands

[no limit] Maximum allowable delay of commands and set point commands

TRANSMISSION OF INTEGRATED TOTALS

(station-specific parameter, mark 'X' if function is only used in the standard direction, 'R' if only used in the reverse direction, and 'B' if used in both directions).

- ☒ Mode A: Local freeze with spontaneous transmission
- ☒ Mode B: Local freeze with counter interrogation
- ☒ Mode C: Freeze and transmit by counter-interrogation commands
- ☒ Mode D: Freeze by counter-interrogation command, frozen values reported spontaneously
- ☒ Counter read
- ☒ Counter freeze without reset
- ☐ Counter freeze with reset
- ☐ Counter reset
- ☒ General request counter
- ☐ Request counter group 1
- ☐ Request counter group 2
- ☐ Request counter group 3
- ☐ Request counter group 4

PARAMETER LOADING

(station-specific parameter, mark 'X' if function is only used in the standard direction, 'R' if only used in the reverse direction, and 'B' if used in both directions).

- ☐ Threshold value
- ☐ Smoothing factor
- ☐ Low limit for transmission of measured values
- ☐ High limit for transmission of measured values

PARAMETER ACTIVATION

(station-specific parameter, mark 'X' if function is only used in the standard direction, 'R' if only used in the reverse direction, and 'B' if used in both directions).

- ☐ Act/deact of persistent cyclic or periodic transmission of the addressed object

TEST PROCEDURE

(station-specific parameter, mark 'X' if function is only used in the standard direction, 'R' if only used in the reverse direction, and 'B' if used in both directions).

- ☒ Test procedure

FILE TRANSFER

(station-specific parameter, mark 'X' if function is used).

File transfer in monitor direction

- [X*] Transparent file
- [] Transmission of disturbance data of protection equipment
- [X] Transmission of sequences of events
- [] Transmission of sequences of recorded analogue values

* a data can be transparently transported by the system but not generated or evaluated. Maximum file size is 16711680 bytes.

File transfer in control direction

- [X] Transparent file

BACKGROUND SCAN

(station-specific parameter, mark 'X' if function is only used in the standard direction, 'R' if only used in the reverse direction, and 'B' if used in both directions).

- [X] Background scan

ACQUISITION OF TRANSMISSION DELAY

(station-specific parameter, mark 'X' if function is only used in the standard direction, 'R' if only used in the reverse direction, and 'B' if used in both directions).

- [] Acquisition of transmission delay

DEFINITION OF TIME OUTS: 104 ONLY

Parameter	Default value	Remarks	Selected value
t0	30 s	Time-out of connection establishment	setting
t1	15 s	Time-out of send or test APDUs	setting
t2	10 s	Time-out for acknowledges in case of no data messages; t2 < t1	setting
t3	20 s	Time-out for sending test frames in case of a long idle state; t3 > t1	setting

Recommended range for timeouts t0 - t2 : 1s to 255s, accuracy 1s

Recommended range for timeout t3: 0s to 48hrs, accuracy 1s

Long timeouts for t3 may be needed in special cases where satellite links or dialup connections are used (e.g. to establish connection and collect values only once per day or week). For dialup connections it may be necessary to give up the connection supervision completely. This is achievable by setting the timeout t3 to zero.

MAXIMUM NUMBER OF OUTSTANDING I FORMAT APDUS K AND LATEST ACKNOWLEDGE APDUS (W): 104 ONLY

Parameter	Default value	Remarks	Selected value
k	12 APDUs	Maximum difference receive sequence number to send state variable	setting
w	8 APDUs	Latest acknowledge after receiving w I format APDUs	setting

Recommended range of values k: 1 to 32767 APDUs, accuracy 1 APDU

Recommended range of values w: 1 to 32767 APDUs, accuracy 1 APDU (Recommendation: w should not exceed two-thirds of k)

PORTNUMBER: 104 ONLY

Parameter	Default value	Remarks
Portnumber	2404	setting

REDUNDANT CONNECTIONS

[] Number N of redundancy group connections used

RFC 2200 SUITE

RFC 2200 is an official Internet Standard which describes the state of standardization of protocols used in the Internet as determined by the Internet Architecture Board (IAB). It offers a broad spectrum of actual standards used in the Internet. The suitable selection of documents from RFC 2200 defined in this standard for given projects has to be chosen by the user of this standard.

- [] Ethernet 802.3
- [] Serial X.21 interface
- [] Other selection from RFC 2200:

List of valid documents from RFC 2200

1.
2.
3.
4.
5.
6.
7. etc.

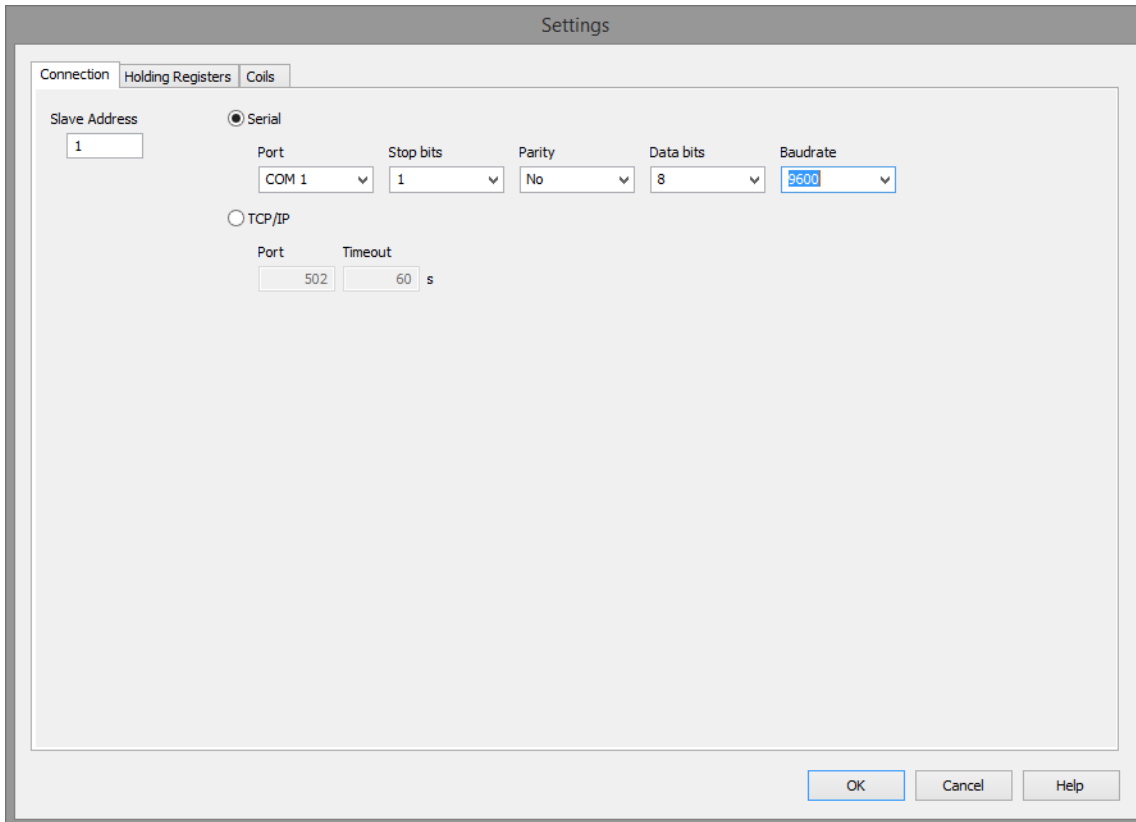
11. MODBUS Slave

The **Process Gateway** for **MODBUS Slave** supports:

- ▶ Function Code 1:
Read Coils
- ▶ Function Code 3:
Read Multiple Registers
- ▶ Function Code 5:
Write Single Coil
- ▶ Function Code 6:
Write single Register
- ▶ Function Code 15:
Write Multiple Coils
- ▶ Function Code 16:
Write Multiple Registers

11.1 Module configuration

Clicking on the Settings button in the start dialog for the Process Gateway opens the dialog for selecting the variables. The variables selected in this way are continuously synchronized in zenon or in the MODBUS system.



The image shows a 'Settings' dialog box with three tabs: 'Connection', 'Holding Registers', and 'Coils'. The 'Connection' tab is active. It contains two main sections: 'Serial' and 'TCP/IP'. The 'Serial' section is selected with a radio button. It includes a 'Slave Address' field with the value '1'. Below it, there are five dropdown menus: 'Port' (COM 1), 'Stop bits' (1), 'Parity' (No), 'Data bits' (8), and 'Baudrate' (9600). The 'TCP/IP' section is unselected and includes a 'Port' field with the value '502' and a 'Timeout' field with the value '60 s'. At the bottom right of the dialog are three buttons: 'OK', 'Cancel', and 'Help'.

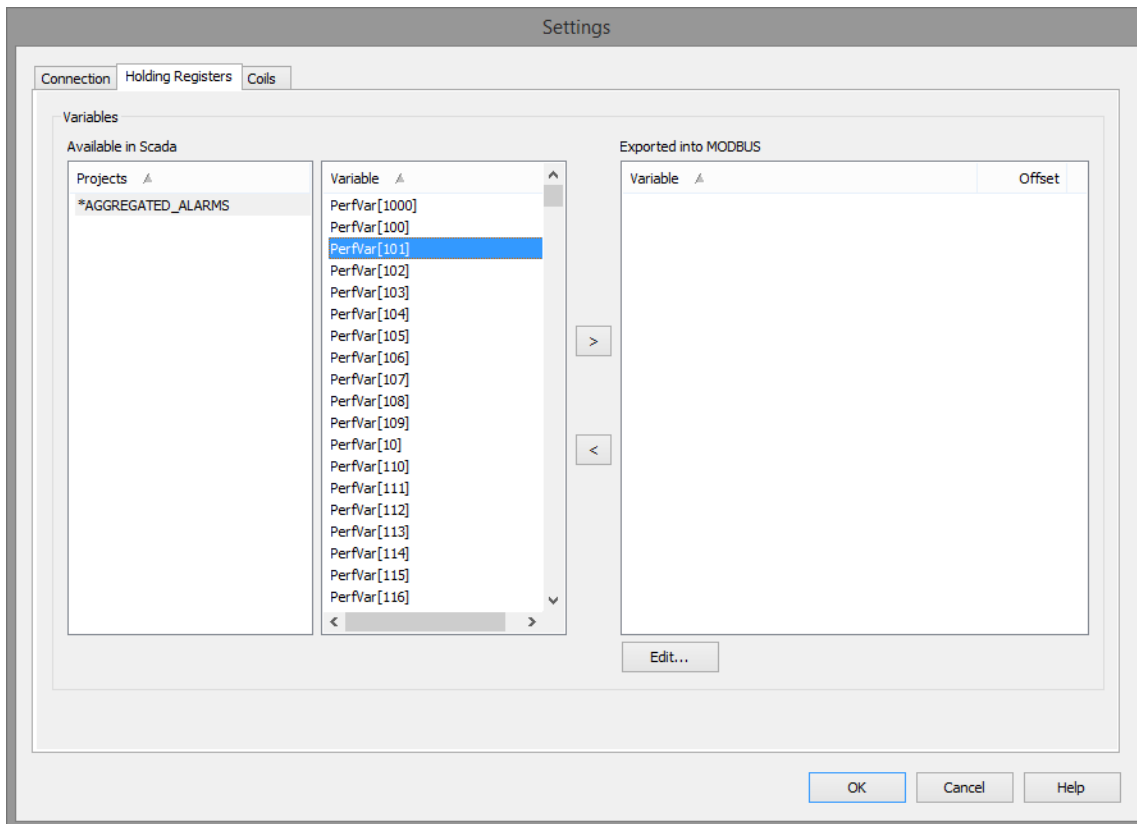
Field	Value
Slave Address	1
Port (Serial)	COM 1
Stop bits (Serial)	1
Parity (Serial)	No
Data bits (Serial)	8
Baudrate (Serial)	9600
Port (TCP/IP)	502
Timeout (TCP/IP)	60 s

MODBUS CONNECTION

MODBUS Connection	MODBUS connection settings.
Slave Address Unit Identifier	MODBUS hardware address for the MODBUS communication. <ul style="list-style-type: none"> ▶ Slave Address for serial communication. ▶ Unit Identifier for TCP/IP communication. Default: 1
Serial	Active: Serial connection is used. <ul style="list-style-type: none"> ▶ Port: Serial interface that is used for access to the MODBUS system. Select from drop-down list. ▶ Stop bits: Number of stop bits. Select from drop-down list. Default 1 ▶ Parity: Parity setting. Selection from drop-down list: No (Default) Odd Even ▶ Data bit: Number of data bits. Select from drop-down list. Default 8 ▶ Baud rate: Baud rate of the serial interface. Select from drop-down list. Default 9600
TCP/IP	Communication to Master via TCP/IP protocol <ul style="list-style-type: none"> ▶ Active: TCP/IP connection with MODBUS TCP is used. ▶ Inactive: serial connection with MODBUS RTU ▶ Port: Port address for the connection. ▶ Timeout: Waiting period.
OK	Applies settings and closes the dialog.
Cancel	Discards all changes and closes the dialog.

11.2 Holding Registers

Clicking on the Settings button in the start dialog for the Process Gateway opens the dialog for selecting the variables. The variables selected in this way are continuously synchronized in zenon or in the MODBUS system.



VARIABLES

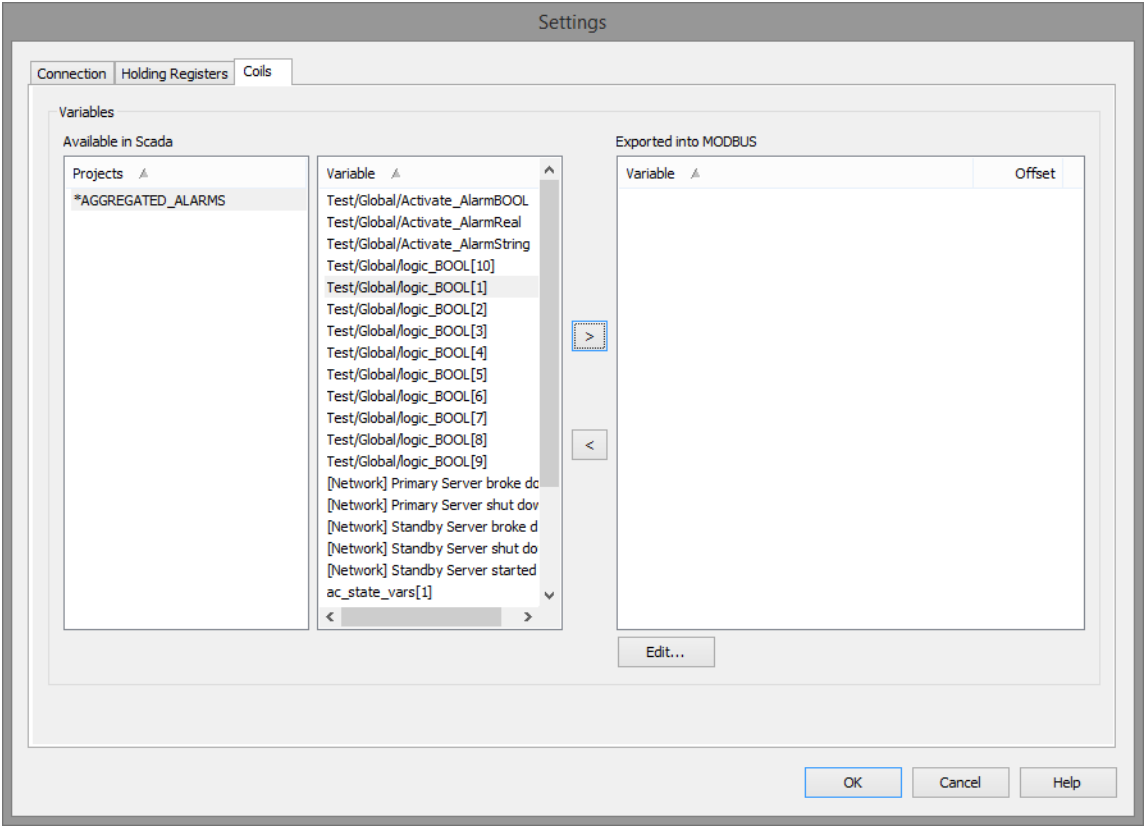
Selection of variables that are exported to MODBUS and synchronized with zenon.

Parameter	Description
Available in Scada	<p>Displays all variables available in zenon</p> <p>With multi-project administration, variables from active projects can be selected.</p> <ul style="list-style-type: none"> ▶ Projects: shows all available projects; the standard project is marked with a * ▶ Variables: offers all variables from the selected project to be transferred
Exported into MODBUS	<p>Lists all variables exported to MODBUS and their offset. List can be sorted by clicking on headings.</p> <ul style="list-style-type: none"> ▶ Variables: List of all SCADA variables that are exported to the MODBUS. Variables that do not come from the standard project receive the project name as a prefix, separated by a # in front of the variable name. ▶ Offset: Offset of the variable (MODBUS holding register offset)
Pfeil-Tasten	<p>Allocation of the variables to MODBUS:</p> <ul style="list-style-type: none"> ▶ Clicking on the ► button adds variables from the Available in Scada list to the Exported into MODBUS list. A dialog to allocate the address is opened (see Allocation dialog section). ▶ Clicking on the ◀ button removes variables from the Exported into MODBUS list and inserts them back into the Available in Scada list.
Edit...	<ul style="list-style-type: none"> ▶ Opens the dialog for addressing (on page 257) of the selected zenon variable to an offset MODBUS.

Note:

- ▶ **Change of name of project or variables in the Editor:**
If a project name or a variable name is changed in the Editor, this change must subsequently be carried out in the configuration file (on page 258), so that the name in the INI file and in the prefix of the **Exported into MODBUS** are amended accordingly.
- ▶ **Change to standard project:**
If the standard project is changed in multi-project administration, the variables must be amended.
Variables of the standard project must not have a prefix. All other variables need a **Projectname#** prefix.
- ▶ **Write Holding Register:**
 - Maximum **block size**: 123

11.3 Coils



VARIABLES

Selection of variables that are exported to MODBUS and synchronized with zenon.

Parameter	Description
Available in Scada	<p>Displays all variables available in zenon</p> <p>With multi-project administration, variables from active projects can be selected.</p> <ul style="list-style-type: none"> ▶ Projects: shows all available projects; the standard project is marked with a * ▶ Variables: offers all variables from the selected project to be transferred
Exported into MODBUS	<p>Lists all variables exported to MODBUS of data type <code>BOOL</code> and their offset. List can be sorted by clicking on headings.</p> <ul style="list-style-type: none"> ▶ Variables: List of all SCADA variables that are exported to the MODBUS. Variables that do not come from the standard project receive the project name as a prefix, separated by a # in front of the variable name. ▶ Offset: Offset of the variable (MODBUS holding register offset)
Pfeil-Tasten	<p>Allocation of the variables to MODBUS:</p> <ul style="list-style-type: none"> ▶ Clicking on the ► button adds variables from the Available in Scada list to the Exported into MODBUS list. A dialog to allocate the address is opened (see Allocation dialog section). ▶ Clicking on the ◀ button removes variables from the Exported into MODBUS list and inserts them back into the Available in Scada list.
Edit	<p>Opens the dialog for addressing (on page 257) of the selected zenon variable to an offset MODBUS.</p>

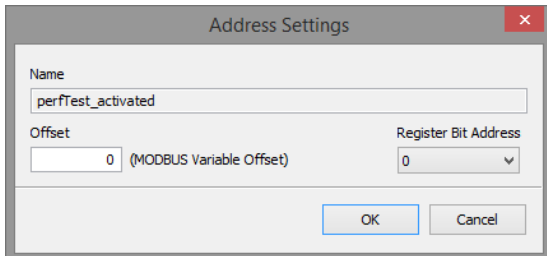
Note:

- ▶ **Change of name of project or variables in the Editor:**
If a project name or a variable name is changed in the Editor, this change must subsequently be carried out in the configuration file (on page 258), so that the name in the INI file and in the prefix of the **Exported into MODBUS** are amended accordingly.
- ▶ **Change to standard project:**
If the standard project is changed in multi-project administration, the variables must be amended.
Variables of the standard project must not have a prefix. All other variables need a **Projectname#** prefix.
- ▶ **Write Holding Register:**
 - Maximum **block size**: 123

11.4 Address Settings

ALLOCATION DIALOG

If a variable is moved to the `Exported into MODBUS` list via clicking the button **>**, a dialog for the name assignment of the zenon variable to an offset in MODBUS opens:



The dialog box titled "Address Settings" contains the following fields and buttons:

- Name:** A text field containing "perfTest_activated".
- Offset:** A text field containing "0" with the label "(MODBUS Variable Offset)" next to it.
- Register Bit Address:** A dropdown menu showing "0".
- Buttons:** "OK" and "Cancel" buttons at the bottom right.

Parameter	Description
Name	Name of the variable. Note: The name of the variable is grayed out, because it can no longer be changed.
Offset	Offset: MODBUS holding register offset. Offset of the variable in MODBUS Attention: 0 is always entered as the offset by default.
Register Bit Address	<ul style="list-style-type: none"> ▶ Select from drop-down list: Bool: Bit-number in the Holding Register Default: 0 ▶ USINT: Low Byte oder High Byte the Holding Register. Default: High Byte Note: not available for the assignment of Coils .
OK	Applies settings and closes the dialog.
Cancel	Discards all changes and closes the dialog.

11.5 Hardware address of the MODBUS system

The MODBUS system of the **Process Gateway** has a MODBUS hardware address that can be defined. This is set in the configuration dialog (on page 251) with the **Slave Address property**. The address 1 is given as standard.

11.6 Configuration file: specific entries for AccessMODBUS

The configuration file `zenProcGateway.ini` contains specific entries for MODBUS. The file must be in the system folder.

[MODBUS]

Entry	Description
COMPORT=	serial interface (COM1 = 0, COM2 = 1,...). Can be set using the configuration dialog (on page 251).
BAUD=	Baud rate of the serial interface. Can be set using the configuration dialog (on page 251).
BYTESIZE=	number of data bits of the serial interface
PARITY=	Parity settings of the serial interface: 0=No 1=Odd 2=Even Can be set using the configuration dialog (on page 251).
STOPBITS=	Number of stop bits of the serial interface 0=1 1=1.5 2=2 Can be set using the configuration dialog (on page 251).
TIMEOUT=	timeout interval for serial communication in milliseconds
HWADDRESS=	Hardware address of the MODBUS system. Default: 1 Can be set using the configuration dialog (on page 251).
REFRESH RATE=	Update time in milliseconds
SERIELL=	Serial or TCP/IP communication: 1 = serial 0 = TCP/IP Can be set using the configuration dialog (on page 251).
PORT=	TCP/IP port
TCP TIMEOUT=	TCP/IP timeout interval in seconds

[MODBUS VARIABLES]

Variables that are to be replaced in AccessMODBUS.

Note: This entry replaces the original **[VARIABLES]** entry from zenon 7.11 onwards. This prevents unnecessary replacement using its **Logic to SCADA connection** for variables from the standard project.

Entry	Description
COUNT	number of variables to be exported
OFFSET_n	Modbus address that is allocated to the corresponding number (n). Numbering starts with 0.
NAME_n	<p>Name of the variable that is allocated to the corresponding number (n). Numbering starts with 0.</p> <p>Examples:</p> <ul style="list-style-type: none"> ▶ OFFSET_0=0 NAME_0=EMS_Supply area 1_forecast_final_consumption Variable from start project ▶ OFFSET_1=2 NAME_1=BASISTUTORIAL#Temperaturfühler Variable from projekt "BASISTUTORIAL"

▶

12. MS Azure

The **Process Gateway** uses the **AccessAzure.dll** to establish a cyclical connection to MS Azure. Services in MS Azure supported by Process Gateway:

- ▶ Servicebus Queue
- ▶ Event Hub
- ▶ IoT Hub

In addition, the Process Gateway can receive data from the IoT hub via `Cloud2Device-Message`.

Data from the **Queue** can then be obtained with the **AzureDrv** driver from MS Azure and integrated into zenon processes. **IoT Hub** content can only be read by the driver if it is transferred to the **Servicebus Queue**. This can be implemented with the standard services of the MS Azure Cloud.

Data from an **Event Hub** is for third-party applications and can no longer be read by zenon.

DRIVER COMMUNICATION

To read the values from the Servicebus Queue, the **AzureDrv** driver establishes a connection to the service bus Queue with the configured name and takes all messages received from it. These messages are unpacked and the online values contained therein are allocated to the variables. The key for this is the **Symbolic address**.

All messages that are already in the queue when the connection is first successfully established are loaded and discarded. It is always only the current values that are displayed.

This means:

Each driver instance on each computer has its own **Queue** as an input signal.



Example

The Servicebus Queue for the computer **MYSERVER1** and the configured prefix **onlinedata** is called the following in MS Azure: **onlinedata_myserver1**.

Numeric (**DOUBLE**) and alphanumeric (**STRING**) values are supported. The time stamp and the system status bits are transferred to the target variable.

You can also find general information on MS Azure in the **MS Azure** manual.

12.1 Module configuration

Configuration dialog for MS Azure connection via Process Gateway.

VARIABLES

In the **Variables** group, you configure the variables whose values are saved by zenon in an MS Azure Service Bus.

Parameter	Description
Available in Scada	Displays all variables available in zenon With multi-project administration, variables from active projects can be selected.
Projects	List of all available projects. The standard project is marked with a *.
Variables	List of all variables of the selected project. List can be sorted; multiple selection is possible. Hint: Double clicking on the variable moves it.
Button >	Selected variables from the list of variables are moved to the Exported into MS Azure list.
Button <	Selected variables are removed from the Exported into MS Azure list.
Exported into MS Azure	<p>List of the variables that are written to the MS Azure Service Bus by the Process Gateway. These are displayed with name (Points) and communication Direction. Double click on an entry to reverse the direction of communication.</p> <p>The key for the values in MS Azure is always PROJECTNAME#VARIABLENAME.</p> <p>Points: The name consists of:</p> <ul style="list-style-type: none"> ▶ Project name ▶ # (as separator) ▶ Variable name. <p>Direction: Shows the direction of communication:</p> <ul style="list-style-type: none"> ▶ read only ▶ write only ▶ read/write <p>Default: read only Change the direction by double-clicking on the variable.</p> <p>Note: Only linked variables can be described.</p>

MS AZURE CONNECTIONS

All target connections in which the current values of the selected variable are to be inserted in MS Azure are to be entered into the **MS Azure Connections** group.

In doing so, the current variable values are added to the connection created in *All* at the same time.

Parameter	Description
List of MS Azure connections	<p>Lists all connections to MS Azure configured. Each connection consists of the connection name (MS Azure Connection) and the ServiceBus Queue Name.</p> <ul style="list-style-type: none"> ► Creation of a new connection with the button New... ► A selected connection can be amended with the Edit... button.
MS Azure Connection	MS Azure Connection address
Service Bus Queue Name	Name of the queue in the MS Azure service bus.
New...	Opens dialog to configure the MS Azure connection.
Edit...	Opens existing connections to configure the MS Azure connection.
Delete	Deletes the selected MS Azure connection from the list.

MS AZURE SETTINGS

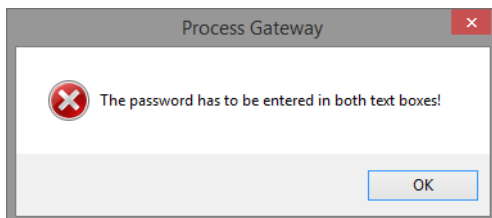
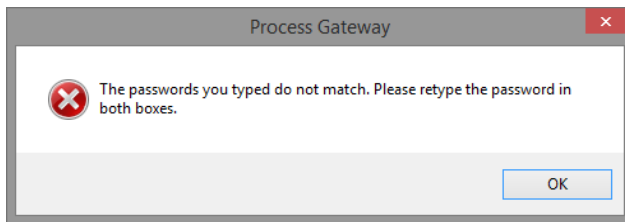
Parameter	Description
Integrity period	<p>Time interval in which the current values of the selected variables are written as an image to the MS Azure Service Bus queue.</p> <p>If the value of a variable changes during this interval, the value change is immediately transferred to the MS Azure queue.</p> <p>Default: 5 s</p>
HTTP Proxy Domain	Address of the proxy server in the network.
HTTP Proxy User	Input field of the user name for login on the proxy server.
HTTP Proxy Password	<p>Input field for password for login on the proxy server.</p> <p>Note: Input is shown with dots - even during entry.</p>
Confirm HTTP Proxy Password	<p>Input field for the confirmation of the password for login on the proxy server.</p> <p>Note: Input is shown with dots - even during entry.</p>

NAVIGATION

Parameter	Description
OK	Applies settings and closes the dialog.
Cancel	Discards all changes and closes the dialog.

DIALOG IN THE EVENT OF INCORRECT ENTRIES

Configurations of the proxy server are validated. A corresponding warning dialog is shown in the event of an incorrect configuration.

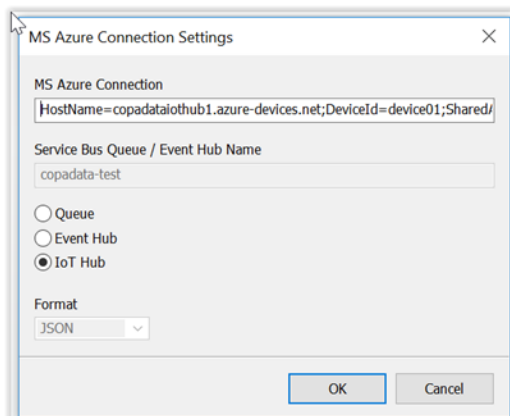


Attention

.NET 4.5 must be installed on the computer in order for the proxy settings to work.

12.1.1 MS Azure Connection Settings

Configuration dialog for the connection to **MS Azure**:



Parameter	Description
MS Azure Connection	<p>MS Azure Connection address</p> <p>Note: You can read and copy this address in the Azure administration portal under Manage Connection Strings of the desired Servicebus Namespace.</p>
Service Bus Queue / Event Hub Name	<p>Name of the MS Azure Service Bus queue or the Event Hub. Selection by means of radio buttons.</p> <p>Note: Only characters that appear in the name of an MS Azure Service Bus Queue or in an Event Hub are permitted. Use simple, short and meaningful names. Avoid special characters, language-specific letters and blank spaces.</p> <p>Service Bus Queue</p> <p>The name of the Service Bus Queue comprises:</p> <ul style="list-style-type: none"> ▶ A freely-configurable prefix ▶ An underscore (_) ▶ The NETBIOS computer name (without domain name suffix) in small lettering <p>If the name does not yet exist in the Namespace a Queue with this name is created. Requirement: The corresponding rights are present.</p> <p>Event Hub</p> <ul style="list-style-type: none"> ▶ Message format: Select in the drop-down list in the Format option. ▶ Event Hub Name: Name of the Event Hub in the MS Azure Service Bus The Event Hub must already be created with this name in the Namespace. <p>Note: Grayed out if the IoT Hub option is activated as the type of communication to MS Azure.</p>
Queue	<p>Communication to MS Azure via <code>Service Bus Queue</code>.</p> <ul style="list-style-type: none"> ▶ Active: The connection is established using the name of the Service Bus Queue.
Event Hub	<ul style="list-style-type: none"> ▶ Communication to MS Azure via <code>Event Hub</code>. ▶ Active: The connection is established using the name of the Event Hub. Selection of the message format using the Format option.

Parameter	Description
IoT Hub	<ul style="list-style-type: none"> ▶ Communication to MS Azure via IoT Hub. ▶ Active: Sending of actual values and the receipt of messages via IoT Hub. Entry of the Device-Connection Strings in the MS Azure Connection input field.
Format	<p>Message format for connection via Event Hub. Select from drop-down list:</p> <ul style="list-style-type: none"> ▶ XML ▶ JSON ▶ BOND (compact binary)
OK	Applies settings and closes the dialog.
Cancel	Discards all changes and closes the dialog.



Information

*The Service Bus Queue Name can be freely configured.
This queue is automatically created in MS Azure during the first communication to MS Azure.*

For each computer that calls up data from MS Azure, use the **AzureDrv** driver to create a separate MS Azure Connection.

Communication via IoT hub

The following is applicable for communication via IoT hub:

- ▶ All content is transferred in the correct format using the **Process Gateway**.
- ▶ Actual values are sent to the **IoT Hub** as a message.
- ▶ Messages received by the **IoT Hub** are decoded and written to the contained value as a set value.

IOT HUB MESSAGE

A message via IoT Hub contains the following content:

Parameter	Data Type	Read (receive)	Write (send)	Description
project	STRING	X	X	zenon project name
variable	STRING	X	X	zenon variable name
isstring	NUMBER	X	X	Type of the zenon variable <ul style="list-style-type: none"> ▶ 0 = Variable value is numerical ▶ 1 = Variable value is a string
numvalue	NUMBER	X	X	Numerical value of the zenon variable. Note: only for numerical values if isstring=0. For string variables, the value is for numvalue 0.
strvalue	STRING	X	X	String value Note: only for string variables if isstring=1. The value for strvalue is empty for numerical values.
time	ISO timestamp	X	X	Time stamp of the zenon variable.
status	NUMBER	X	X	Status of the zenon variable.
isintegrity	NUMBER	--	X	Trigger for data transfer. <ul style="list-style-type: none"> ▶ 0 = Message sent to MS Azure due to a value change. ▶ 1 = Message was sent cyclically. Note: Cyclical sending corresponds to the configured Integrity period option in the module configuration (on page 261).
sequence	NUMBER	--	X	Serial number for sent messages. This is automatically carried out by the Process Gateway and is increased for each message.
msgtime	ISO timestamp	--	X	Timestamp of the message.

Key:

x: supported

--: not supported

The messages in receive direction are identical to those in send direction. The `isintegrity`, `sequence` and `msgtime` parameters are not needed to receive. All content is transferred in the correct format using the **Process Gateway**. All content is transferred in the correct format using the **Process Gateway**.



Example

Possible format of a message that is sent by Process Gateway:

```
{"isintegrity":1,"isstring":1,"msgtime":"2017-05-03T08:51:23.260Z","numvalue":0,"project":"AZURE_TEST","sequence":1,"status":1078067200,"strvalue":"sunshine","time":"2017-04-11T05:28:33.542Z","variable":"weather"}
```

12.2 Message formats

SERVICE BUS QUEUE

The messages in the Queue must have a serialized .NET class `OnlineValueMessage`.

FORMAT:

```
public class OnlineValueMessage
{
    public string strVarProject { get; set; }
    public string strVarName { get; set; }
    public long nVarID { get; set; }
    public double fValue { get; set; }
    public string strValue { get; set; }
    public bool bValueIsString { get; set; }
    public long nTime_s { get; set; }
    public short nTime_milli { get; set; }
    public long nStatus { get; set; }
}
```

EVENT HUB

When using the Event Hub as a connection, it is possible to choose between XML, JSON and BOND (compact binary) as a message format. Partition-Key is the zenon variable name.

FORMAT BOND

```
namespace AzureServiceBusShared
```

```
struct ArchiveMessage
{
    0: string strArvProject;
    1: string strArvName;
    2: string strVarProject;
    3: string strVarName;
    4: int64 nVarID;
    5: int16 nCalc;
    6: double fValue;
    7: string strValue;
    8: bool bValueIsString;
    9: int64 nTime_s;
    10: int16 nTime_milli;
    11: int64 nStatus;
}
```

```
struct OnlineValueMessage
{
    0: string strVarProject;
    1: string strVarName;
    2: int64 nVarID;
    3: double fValue;
    4: string strValue;
    5: bool bValueIsString;
    6: int64 nTime_s;
    7: int16 nTime_milli;
    8: int64 nStatus;
}
```

12.3 Error message

Error message in the LOG file:

Error message	Debug Level	Description
Project %s inactive fail!	Error	The project-inactive event could not be processed by the Gateway in the given time.
Project %s active fail!	Error	The project-active event could not be processed by the Gateway in the given time.
Value change queue full: Value change for id:%u of project:%s lost!	Error	The queue for the value changes from Runtime is full. Additional values are discarded. The connection to Azure is probably too slow in order to forward all value changes.
Adding of %d Online Rows failed for '%s' Queue '%s'	Error	The insertion of value changes into the service bus queue was unsuccessful.
Adding of %d Online Rows succeeded for '%s' Queue '%s'	Deep Debug	The insertion of value changes into the service bus queue was successful.
Starting to add Online Rows failed for '%s' Queue '%s'	Error	The establishment of the a connection to the service bus queue was unsuccessful.
Creating Azure Wrapper failed	Error	The ManagedAzureWrapper.dll could not be loaded.

IOT

Parameter	Description	Description
Sending value to '%s' failed	Error	The writing of the set value has failed
IoTHubMessage_GetByteArray() failed	Error	The SDK API call has failed.
Received message from IoT Hub	Deep Debug	Message from IoT hub received.
Exception when deserializing payload: %s	Error	Received message is in the wrong format
IoTHubClient_CreateFromConnectionString() failed	Error	The SDK API call has failed.
IoTHubClient_SetMessageCallback() failed	Error	The SDK API call has failed.
IoTHubMessage_CreateFromByteArray() failed	Error	The SDK API call has failed.
Sent message to IoT Hub	Deep Debug	The message was sent.

12.4 Configuration file: specific entries for MS Azure

The configuration file `zenProcGateway.ini` contains this entries for MS Azure.

[AZURE]

Entry	Description
INTEGRITYPERIOD=	Interval of the write cycle in seconds.
PROXY_DOMAIN=	Address of the proxy server.
PROXY_USER=	User name for login on the proxy server
PROXY_PWD=	Password for login on the proxy server. Note: The password is also shown in encrypted form in the .INI file.

Note: .NET 4.5 must be installed on the computer in order for the proxy settings to work.

[VARIABLES]

Entry	Description
Name_n=	Name of the variable for Process Gateway. Format: SOURCEPROJECT#SOURCEVARIABLE. Numbering (n): n Stands for the serial number of the connection configuration. This numbering starts with 0. Example: <ul style="list-style-type: none"> ▶ 1 variable results in n = 0 ▶ 3 variables result in the following for the third entry: n = 2
Count=	Number of variables. Note: The numbering starts with 1 here. Eight connections result in Count=8.

[CONNECTIONS]

Entry	Description
CONN_n=	<p>Name of the MS Azure connection.</p> <p>Numbering (n):</p> <p>n Stands for the serial number of the connection configuration.</p> <p>This numbering starts with 0.</p> <p>Example:</p> <ul style="list-style-type: none"> ▶ 1 connection results in n = 0 ▶ 3 connections result in the following for the third entry n = 2
QUEUE_n=	<p>Name of the MS Azure queue.</p> <p>For the numbering (_n) the same rules apply as for CONN</p>
COUNT=	<p>Number of the configured MS Azure connections.</p> <p>Note: The numbering starts with 1 here. Two connections result in Count=2.</p>

13. OPC UA Server

The **OPC UA Server** provides the variables used in the project to all standard-compliant **OPC UA Client** tools.



Information

*The **OPC UA Server** can be used for all zenon versions from 5.50.*

The predecessor to the **OPC UA Server** is the OPC DA server. This is not part of **Process Gateway**.

The **OPC UA driver** and all dialogs are only available in English.

MULTI-PROJECT CAPABILITY

The **OPC UA Server** is multi-project compatible from zenon 7.11 onwards. Variable from the Runtime project and all its subprojects can be selected. In doing so, the object name from the variable name and the project name are combined. Configurations for **OPC UA Clients** that were created before zenon 7.11 are thus not compatible. These must be amended when using zenon 7.11 or higher.

From version 7.60, the `NodeIDs` change if variables are not explicitly defined in the configuration. The data type of the `NodeID` is `string`. The `NodeID` always consists of the project GUID and the variable ID. If variables have been explicitly configured, this change has no effect.



Example

`NodeID: "6d5ea886-1b04-4d9c-aa35-2b5664c500cd-38d"`; the following is applicable in the process: `"6d5ea886-1b04-4d9c-aa35-2b5664c500cd"` is the **Project ID** and `"38d"` corresponds to the variable ID 909.

From version 7.60, the `Namespace Index` is "2" by default, instead of "80", for new configurations. The `Namespace Index` can no longer be configured in the user interface. This change has no effect for existing configurations.

The earlier action can be activated again by means of the `"configuration_version=1"` entry in the INI file. This can, for example, make sense for a `Namespace Index` with 80 or for a configurable `Namespace Index`.

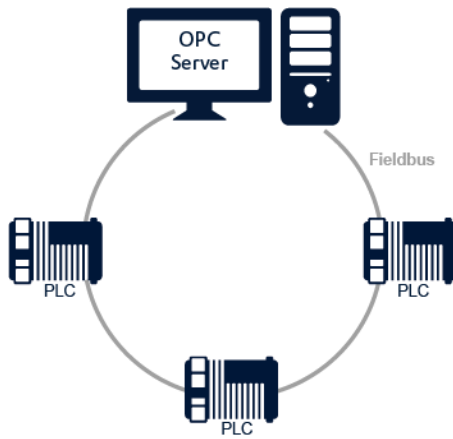
13.1 Compare OPC Server and OPC UA Server

The **OPC UA Server** system has superseded the OPC server. Use of an **OPC UA Server** is recommended for most applications.

OPC SERVER

- ▶ OPC Task Force since 1995
- ▶ Uniform interface for automation systems
- ▶ Based on Microsoft's COM/DCOM technology
- ▶ OPC foundation since 1996

- several specifications for different applications



OPC UA SERVER



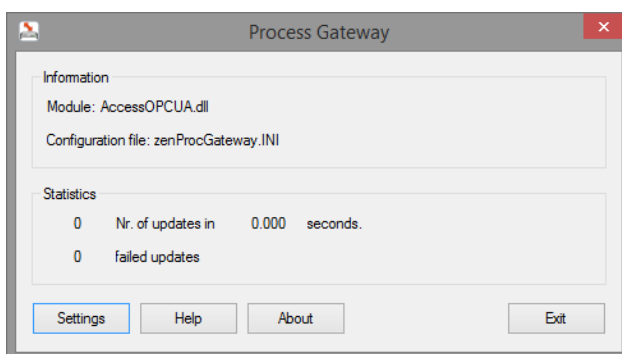
COMPARISON OF OPC SERVER TO OPC UA SERVER

Parameter	OPC Server	OPC UA Server
Data model	Limited	for all applications
Implementation	Manufacturer-dependent	Manufacturer-independent
Interoperability	--	X
Configuration	laborious	simple
Several instances	--	X
Network use	--	X
Network technology	--	OPC UA TCP binary Protocol
Performance	low	high
Platforms	Windows only	independent
Resource requirements	high	lower
Service-orientated architecture	--	X
Security	outdated	In accordance with current standards
Connection security	--	X
Windows CE	--	X

Key:

- ▶ **X**: supported
- ▶ --: not supported

13.2 Module configuration

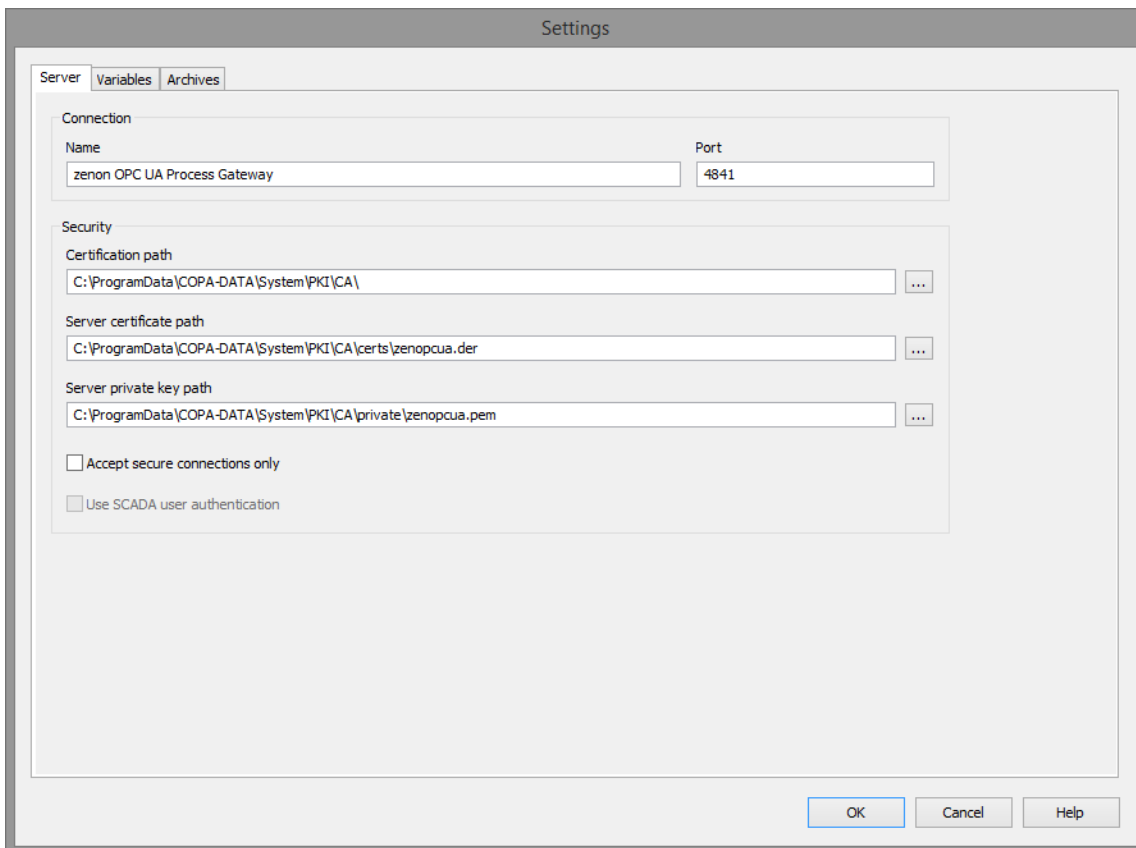


To configure the server and variables for OPC UA:

1. In the **Process Gateway** statistics dialog, click the **Configuration** button.
The dialog for the configuration of the **OPC UA module** is opened.
2. In the module configuration dialog, you configure:
 - Server (on page 277)
Connection parameters to the **OPC UA Server**
 - Variables (on page 283)
Variable assignment

13.2.1 Server

You configure connection parameters to an OPC UA server in the **Server** tab.



The screenshot shows the 'Settings' dialog box with the 'Server' tab selected. The 'Connection' section contains fields for 'Name' (zenon OPC UA Process Gateway) and 'Port' (4841). The 'Security' section contains three fields for paths: 'Certification path' (C:\ProgramData\COPA-DATA\System\PKI\CA\), 'Server certificate path' (C:\ProgramData\COPA-DATA\System\PKI\CA\certs\zenopcua.der), and 'Server private key path' (C:\ProgramData\COPA-DATA\System\PKI\CA\private\zenopcua.pem). There are also two checkboxes: 'Accept secure connections only' and 'Use SCADA user authentication', both of which are currently unchecked. At the bottom right, there are 'OK', 'Cancel', and 'Help' buttons.

CONNECTION

Communication parameters to the OPC/OPC UA server.

Parameter	Description
Name	Name of the server as it is shown on the client. Default: zenon OPC UA Process Gateway
Port	TCP port via which the client establishes the connection to the OPC UA server. Default: 4841

SECURITY

Settings and assignment of certificates for secure and encrypted communication.

Parameter	Description
Certificates path	<p>Path for the root folder with the certificate files and the respective subfolders.</p> <p>Click on the ... button to open the file selection dialog</p> <p>Default: %ProgramData%\PKI\CA\</p> <p>Note: If this root folder is changed, the paths for Server certificate path and Server private key path must also be amended.</p>
Server certificate path	<p>File with the publickey of the server. This publickey is used by the client for the encryption of messages.</p> <p>Click on the ... button to open the file selection dialog Files with the following file type are shown for selection:</p> <ul style="list-style-type: none"> ▶ Format: DER ▶ Suffix: .der <p>Default: %ProgramData%\PKI\CA\certs\zenopcua.der</p> <p>Note: This path must be a subfolder of the configured Certificates path. In the event of an error, a warning dialog is visualized and the OK button is grayed out.</p>
Server private key path	<p>File with the privatekey of the server. This privatekey is used by the server to decrypt the messages from the client.</p> <p>Click on the ... button to open the file selection dialog Files with the following file type are shown for selection:</p> <ul style="list-style-type: none"> ▶ Format: PEM ▶ No password <p>Default: %ProgramData%\PKI\CA\private\zenopcua.pem</p> <p>Note: This path must be a subfolder of the configured Certificates path. In the event of an error, a warning dialog is visualized and the OK button is grayed out.</p>
Accept secure connections only	<p>Checkbox to be ticked if only one secure connection can be used to establish the connection.</p> <ul style="list-style-type: none"> ▶ Active: Establishment of a connection is only accepted with a secure connection. Securing is carried out each time with a signature or with a signature and encryption. ▶ Inactive: Establishment of a connection is only accepted with a secure

	<p>connection.</p> <p>The following are considered a secure connection:</p> <ul style="list-style-type: none"> ▶ Basic128Rsa15: RSA15 as key wrap algorithm and 128-bit basic as algorithm for the message encryption ▶ Basic256: 256-bit basic as algorithm for message encryption <p>Default: Inactive</p> <p>Note: for connections with a signature, the packages are secured against tampering, but the contents of the package are not encrypted.</p>
Trust all client certificates	<p>Checkbox to select whether all client certificates are accepted for the establishment of a connection or only client certificates known on the server.</p> <ul style="list-style-type: none"> ▶ Active: All client certificates are accepted and the connection is made. ▶ Inactive: All clients that want to connect with the server must send a valid certificate. This certificate is also checked to see if it is in the list of known client certificates Certificate trust list path. <p>Default: active</p> <p>Note: An OPC UA Client without a secure connection does not send a certificate. For an OPC UA Client with a secure connection, the certificate must always be stored in the Trust List folder.</p>
Use SCADA user authentication	<p>Checkbox to configure the connection.</p> <p>If there is no user administration configured in the zenon project, this checkbox is grayed out.</p> <ul style="list-style-type: none"> ▶ Active: Clients can only log on to the server with a valid user ID and password. Note: To do this, the user administration must be activated and configured in zenon Editor. ▶ Inactive: Anonymous login is possible. <p>Attention: The user name and password are not transferred in encrypted form in the OPC UA telegram. If necessary, use your own users in the project without assigned authorization levels.</p>

CLOSE DIALOG

OK	Applies all changes in all tabs and closes the dialog. <i>Attention:</i> All active connections are ended and the server is restarted with the new settings. Connection errors may occur in the process.
Cancel	Discards all changes in all tabs and closes the dialog.
Help	Opens online help.

ERROR MESSAGE

Error messages are displayed in red font. In doing so, not all errors are shown together. The first error of the configuration is shown each time. It is therefore possible that a new error message is shown after a correction has been made.

Certificates

*All certificates must correspond to the format **X509**.*

CERTIFICATE STORE

*The **OPC UA Server** in the **Process Gateway** uses an OpenSSL Directory certificate store.*

SERVER CERTIFICATE

The server certificate created on initial startup is a self-signed certificate. The certificate contains a public key. In addition, a suitable private RSA key (1024 bit) is generated.

*Although the **OPC UA Gateway** can in principle run with the self-created certificate, it is recommended that a certificate that corresponds to the requirements is created for productive operation. To do this, the self-created certificate can serve as a template for applications that allows creation of the certificates according to user criteria.*



Attention

The filename **zenopcua** is used by both the **OPC UA Server** as well as by the **OPCUA32 Client driver**. If encrypted communication between the **OPCUA32 client driver** and **OPC UA Server** in the **Process Gateway** is configured for tests on the same computer, it is best that the certificate for the **OPCUA32 client driver** is renamed before the certificate is copied to the **Trust List** of the **OPC UA Server**.

*As a URI in the **SubjectAltName** field, the certificate gets the Full Qualified Domainname (FQN) of the computer, the default port number 4841 from the **OPC UA Server** and a DNS entry with the computer name. The URI of the **OPCUA Server** is compiled dynamically and changes if, for example, the port number is changed. The server certificate is not changed in this case. A client that carries out a strict check of the server certificate can then reject the server certificate. It can also be necessary to create your own certificate that contains an entry for the IP address in addition to the DNS entry for the **SubjectAltname**.*

CERTIFICATE TRUST LIST/CERTIFICATE TRUST CHAIN

*All certificates with the suffix **.der** are checked by the **OPC UA Server** and classed as a trustworthy certificate. As an alternative to the self-designated Application Instance **certificate** of an **OPC UA Client**, the public root certificate of the Certificate Authority that was used to sign an Application Instance **certificate** of an **OPC UA Client** can also be stored. The Application Instance **certificate** of the client does not need to be present on the server in this case.*

CERTIFICATE REVOCATION LIST

If an **OPC UA client** uses an Application Instance **certificate** that is signed by a Certificate Authority, the Certificate Revocation List is also checked by the server in addition to the certificate. The **OPC UA Server** checks all files with the suffix **.crl** in the **Crl** subfolder.

Attention: The **.crl** file must be stored in PEM format. Only one single **.crl** file can be present per Certificate Authority.

The connection is denied if:

- ▶ The client certificate is included in the **Revocationlist** **BadCertificateRevoked**
- ▶ The CRL is no longer current (**BadCertificateTimeInvalid**)

If, for a certificate that was signed by a Certificate Authority, there is no **Certificate Revocation List** present, it is assumed that there is no such Revocation List. In this case, the certificate is treated as trustworthy.

Generate new certificate

If necessary, new certificates can be generated when the OPC UA server is started.

To do this:

1. Remove the existing certificates.

There are two possibilities for doing so:

- a) Delete the following folder: `%programdata%\COPA-DATA\system\PKI`.
In doing so, all **Trusted certificates** and **Certificate revocation lists** are removed.
Or:

- b) Delete the following files in order to get all other certificates and lists:

zenopcua.der in the folder `%programdata%\COPA-DATA\system\PKI\CA\certs`

zenopcua.pem in the folder

%programdata%\COPA-DATA\system\PKI\CA\private

zenopcua.crl in the folder %programdata%\COPA-DATA\system\PKI\CA\crl

2. Open the **zenprocgateway.ini** configuration file.
3. Go to the section **[OPCUA]**.
4. Remove the following entries:
 - **CERTIFICATES_PATH=**
 - **SERVER_CERTIFICATE_LOCATION=**
 - **SERVER_PRIVATE_KEY_LOCATION=**
5. Start the OPC UA **Process Gateway**.
6. Confirm the message box by clicking on the **Yes** button to create new certificates.



Attention

If server certificates need to be renewed, each OPC UA client that needs a valid server certificate must receive the renewed certificate.

CREATE THE SERVER CERTIFICATE FROM SCRATCH

The following situations can make it necessary to create the server certificate from scratch:

- ▶ several instances of Process Gateway on the same system with different port numbers
- ▶ a change to the port number
- ▶ a change to the computer name
- ▶ the certificate has expired

13.2.2 Variables

Variables for the **OPC UA Process Gateway** can be selected automatically or individually.

Variables from the current project and its subprojects can be selected. The name on the server consists of the variable names and the project names.



Attention

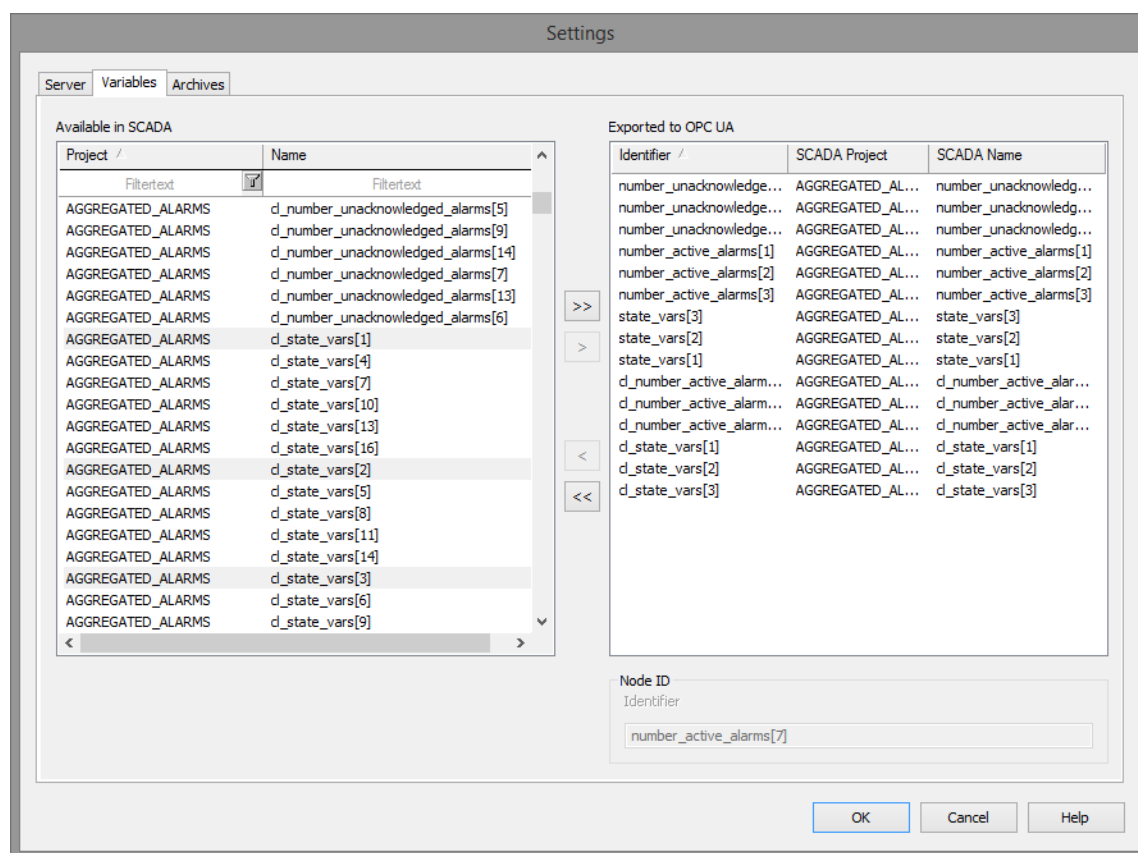
Variables that do not have the **Write set value** property activated in the zenon Editor cannot be written by an **OPC UA Client**.

DEFINING VARIABLES

If no variables are defined in the configuration for the **OPC UA Server**, all variables from the available zenon projects are added to the OPC UA Address Space. All variables (including system driver variables) are visible for the clients.

with manual configuration (at least one configured variable), the values defined by the user are used. The **Namespace Index 2** is used by default and a **String Identifier** with the variable names is entered. The **Namespace Index** cannot be configured.

You configure the variables that are to be provided as a Node in the OPC UA data model in the **Variables** tab.



Parameter	Description
Available in SCADA	<p>List of the configured variables available in the zenon Runtime.</p> <ul style="list-style-type: none"> ▶ Project zenon project name ▶ Name zenon variable name Corresponds to the Name variable property in zenon Editor. <p>The list can be sorted and filtered.</p> <ul style="list-style-type: none"> ▶ Sort by clicking on the column heading. Another click changes the sorting order. ▶ Filter by entering the filter text in the input field. The characters * or ? are possible as placeholders.
Arrow keys	<p>Assignment of the zenon variables to OPC UA:</p> <ul style="list-style-type: none"> ▶ Clicking on the > button adds selected variables from the Available in SCADA list to the Exported to OPC UA list. ▶ Clicking on the < button removes variables from the Exported to OPC UA list and inserts them back into the Available in SCADA list. ▶ Clicking on the >> button adds selected variables from the Available in SCADA list to the Exported to OPC UA list. ▶ Clicking on the << button removes all variables from the Exported to OPC UA list. <p>Multiple selection is possible by clicking the mouse and pressing the CTRL key.</p>
Exported to OPC UA	<p>List of exported variables.</p> <p>The variables can be sorted according to:</p> <ul style="list-style-type: none"> ▶ Identifier ▶ SCADA Project ▶ SCADA name
Node Id	<p>The input field for Node ID is only available if an entry in the Exported to OPC UA list is selected.</p>

NODE ID

Parameter	Description
Identifier	<p>Defines the Node Identifier for the selected OPC UA variable. The Node Identifier type is automatically defined through the entry:</p> <ul style="list-style-type: none"> ▶ A numerical Identifier is automatically created with numerical input. ▶ If a GUID is entered, a GUID Identifier is created automatically, otherwise the Identifier is treated as a string. GUID form: xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx) <p>The field must not be empty.</p> <p>The Node Identifier must be unique in the complete address area.</p>

VARIABLE EXPORT TO OPC UA

To export variables to OPC UA:

1. Select the desired variables and aggregation archives in **Available in SCADA** with a mouse click. Multiple selection is possible.
2. Click on the arrow **Exported to OPC UA**
To export all variables, click on the double arrow

The zenon variable name is used as **Browsename** and **Displayname**.



Information

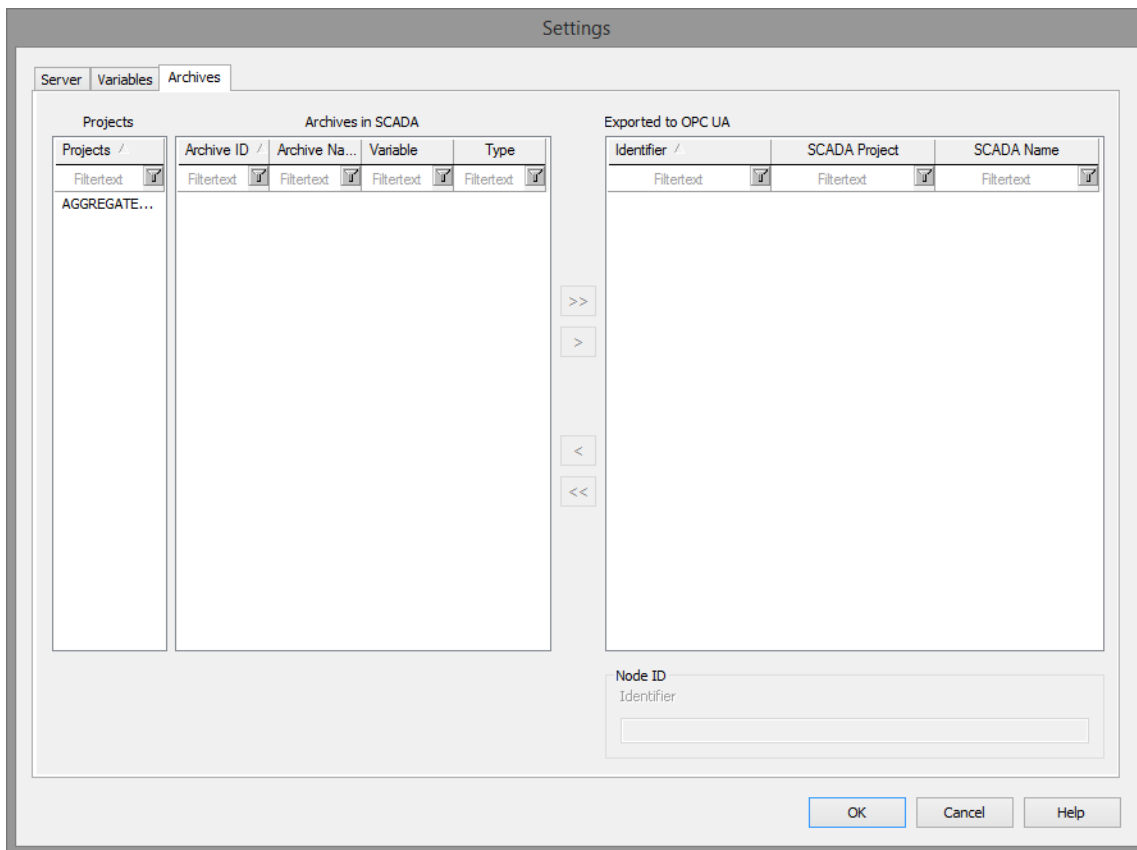
For zenon variables with a slash (/) in the names, the / is replaced by a dot (.). The parts of the name separated by a slash are saved in their own subfolders.

Example: The zenon variable "Static/Scalar/Int32" is divided into:

- ▶ An OPC UA variable "Int32"
- ▶ In a folder called "Scalar",
- ▶ which is a subfolder of the "Static" folder

This works with all data types, multi-dimensional arrays, umlauts in names and slashes at the end.

13.2.3 Archives



PROJECTS

List of the project activated in the zenon Editor.

Parameter	Description
[Project name]	Name of the project loaded in the zenon Editor. In Archives in SCADA the content of the selected project is displayed.

ARCHIVES IN SCADA

List of the archive variable of the selected project.

Parameter	Description
[Variable list]	<p>List of the projected archive variables in the zenon Runtime.</p> <ul style="list-style-type: none"> ▶ Archive ID Identification of the available zenon archive. ▶ Archive Name Projected name of the zenon archive. ▶ Variable Projected name of the archive variable. ▶ Type Archive type <p>The list can be sorted and filtered.</p> <ul style="list-style-type: none"> ▶ Sort by clicking on the column heading. Another click changes the sorting order. ▶ Filter by entering the filter text in the input field. The characters * or ? are possible as placeholders.
Arrow keys	<p>Assignment of the archive variables to OPC UA:</p> <ul style="list-style-type: none"> ▶ Clicking on the ► button adds selected variables from the Archives in SCADA list to the Exported to OPC UA list. ▶ Clicking on the ◀ button removes variables from the Exported to OPC UA list and inserts them back into the Archives in SCADA list. ▶ Clicking on the >> button adds selected variables from the Archives in SCADA list to the Exported to OPC UA list. ▶ Clicking on the << button removes all variables from the Exported to OPC UA list. <p>Multiple selection is possible by clicking the mouse and pressing the CTRL key.</p>
Exported to OPC UA	<p>List of exported variables.</p> <p>The variables can be sorted according to:</p> <ul style="list-style-type: none"> ▶ Identifier - put together from: Project name + archive ID + archive variable name + archive type ▶ SCADA Project Name of the zenon Logic project. ▶ SCADA name -put together from: archive ID + archive variable name +

	archive type
Node ID	The input field for Node ID is only available if an entry in the Exported to OPC UA list is selected.

CLOSE DIALOG

Option	Description
OK	Applies all changes in all tabs and closes the dialog.
Cancel	Discards all changes in all tabs and closes the dialog.
Help	Opens online help.

13.3 Start

When starting, **Process Gateway** loads the configured module.

When first starting or if no variables have been explicitly configured, the **OPC UA Server** automatically creates the data model for all variables from all projects in Runtime. With many variables, or with many variables configured with alarms, the start can take a corresponding length of time. It is, in principle, to be recommended that only the variables that are actually needed by the **OPC UA Client** are provided through the **OPC UA Server**. Several instances of the **OPC UA Process Gateway** can also be started, with a different TCP port and a different configuration in terms of variables.

The driver tries to read the active configuration from the given INI file. If the configuration contains error, or an error occurs on starting, the configuration dialog (on page 276) is opened and an error message is shown with an error text. If the configuration is not confirmed with **OK**, the **Process Gateway** is closed.



Attention

*If the application is started as invisible, there is also no configuration dialog displayed. If the configuration is invalid or not present, the **Process Gateway** is automatically closed.*

13.4 Connection

The **OPC UA Process Gateway** only supports the **OPC UA TCP binary Protocol** and does not contain a **Discovery Server**. The client must also connect directly to the server. The connection address corresponds to the scheme: **opc.tcp://hostname.of.server:port**

The connection can be secured (on page 290).

13.5 Security

The module uses the security procedures that correspond to **OPC UA** (signing and encrypting messages) as defined in the standard. Certificates can be checked or all accepted, depending on the setting (on page 276).

SECURE CONNECTIONS

If the **Accept secure connections only** (on page 277) option is activated, the server only accepts connections that use the **Security Mode** `Sign` or `Sign & Encrypt`. In addition, the **Security Policy** used by the client must be `Basic128Rsa15` or `Basic256`. If this option is deactivated, any desired combination of **Security Mode** and **Security Policy** is accepted.

USER ADMINISTRATION

If user administration is activated in the zenon project and at least one user is created, the option **Use SCADA user authentication** (on page 277) is available. If this option is activated, only clients that send a valid combination of `user` and `password` have access to the **OPC UA Server** data. No check is carried out if this option is inactive. In this case, both anonymous connections and also all user/password combinations are accepted.

13.6 Alarms and Conditions

The **OPC UA Process Gateway** also supports **OPC UA Alarms and Conditions**. When starting, the **OPC UA Server** automatically checks the configured variables for existing limit values or a reaction matrix, with conditions that are defined as an alarm.

An **OPC UA Client** can create a `Subscription for Event Notifications` and add the `Server Object`. The **OPC UA Client** can thus be informed of limit value breaches of a variable with an alarm active. The client is also in a position to confirm the alarm, as in the zenon Runtime.

Attention: The comments for alarms for the **OPC UA** method call for acknowledgment is no longer submitted to zenon Runtime. If an alarm is configured in zenon with "Comment required", an **OPC UA Client** cannot confirm this alarm with a comment.

Attention: If an alarm is configured so that an alarm cause is required, the alarm cannot be acknowledged on the **OPC UA** side.

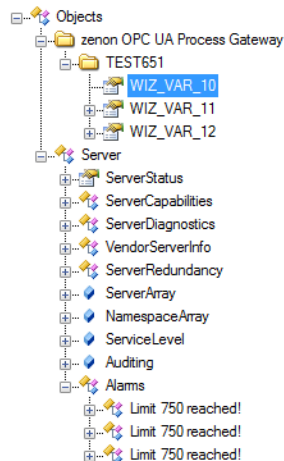
The standard-compliant OPC weighting is shown on the index of the alarm/event classes linked to the limit value. The index of an alarm/event class is automatically incremented by the zenon Editor with each new alarm/event class and can be checked in the properties. Up to 20 classes with the following allocation are supported:

OPC UA range	zenon alarm/event class	OPC weighting
HIGH	20	1000
	19	950
	18	900
	17	850
MEDIUM HIGH	16	800
	15	750
	14	700
	13	650
MEDIUM	12	600
	11	550
	10	500
	9	450
MEDIUM LOW	8	400
	7	350
	6	300
	5	250
LOW	4	200
	3	150
	2	100
	1	50

If a variable with no alarm/event class is linked, the lowest possible classification 1 (LOW) is used.

ILLUSTRATION ZENON ALARM AS OBJECT

zenon variables and related alarms are illustrated in the **OPC UA Server** as objects.



The illustration shows the inclusion of the zenon project (TEST651) and the zenon variables (WIZ_VAR_10, WIZ_VAR_11, WIZ_VAR_12) as objects in the upper area. Below this are the alarms that have been generated when the limit values are reached (Limit 750 reached! etc.).

ALARM CONDITION NOTE IDENTIFIER

The **Node Identifier** type of the **Alarm Condition** is a **String Type Node ID**.

The following structure is used:

```
<project GUID>-<variable ID hexadecimal>@<variable name>:_<limit value description or "Rema">
```

About the INI entry: `USE_LIMIT_TEXT_FOR_ALARM_NODES=TRUE` in the OPCUA section in the ini file for the **OPC UA Process Gateway**, an alternative structure can be activated where the limit value text is used. This alternative structure is only available for limit values and not for alarms via a reaction matrix. A dynamic limit value text is not supported in doing so. It must be ensured that the same limit value text is not used for several limit value texts for the same variable, because otherwise there is no uniqueness:

```
<project GUID>-<variable ID hexadecimal>@<static limit value text>
```

LOGGING

The driver supports a comprehensive LOG function. These LOG entries can be evaluated with the **Diagnosis Viewer** tool.

13.7 Service Sets

Service Set	Is supported
Discovery Service Set	
FindServers	yes
GetEndpoints	yes
RegisterServer	no
SecureChannel Service Set	
OpenSecureChannel	Yes
CloseSecureChannel	Yes
Session Service Set	
CreateSession	yes
ActivateSession	yes
CloseSession	yes
Cancel	yes
NodeManagement Service Set	
AddNodes	no
AddReferences	no
DeleteNodes	no
DeleteReferences	no
View Service Set	
Browse	yes
BrowseNext	yes
TranslateBrowsePathsToNodeIds	yes
RegisterNodes	yes
UnregisterNodes	yes
Query Service Set	
QueryFirst	no
QueryNext	no
Attribute Service Set	
Read	yes

HistoryRead	yes
Write	yes
HistoryUpdate	no
Method Service Set	
Call	yes
MonitoredItem Service Set	
CreateScreenedItems	yes
ModifyScreenedItems	yes
SetMonitoringMode	yes
SetTriggering	yes
DeleteScreenedItems	yes
Subscription Service Set	
CreateSubscription	yes
ModifySubscription	yes
SetPublishingMode	yes
Publish	yes
Republish	yes
TransferSubscriptions	yes
DeleteSubscriptions	yes

13.8 Communication

SUBSCRIPTIONS

The **OPC UA Server** supports communication by means of Subscriptions. Several Subscriptions with different settings can be created per session by a client. The maximum number of Subscriptions per Session is limited to 10 by default. The "MAX_COUNT_OF_SUBSCRIPTIONS_PER_SESSION" ini entry in the "OPCUA" section can be used to configure the maximum number between "1" and "100".

PUBLISHREQUESTS

A maximum of 10 `PublishRequests` for the queue are permitted by the **OPC UA Server** per session. This depends on the number of `Subscriptions`.

- ▶ **Sequence for publish requests (first in, first out):**
When sending, the incoming sequence of the `PublishRequest` is taken into account. If a `Subscription` is executed and a `PublishResponse` must be sent for a `DataChangeNotification` or a `KeepAlive`, the oldest `PublishRequest` from the queue is always taken.
- ▶ **BadNoSubscription for all publish requests if the last Subscription has been deleted or the session has been closed:**
If the last `Subscription` has been logged off from the server, for `PublishRequests` that are still logged onto the server, `PublishResponse` with the value `OpcUa_BadNoSubscription` is created and sent to the client. The open `PublishResponses` are then deleted on the server.
- ▶ **timeoutHint for PublishRequests:**
Before the `PublishResponses` for the `Subscriptions` are sent, it is ensured that only valid - i.e. not expired `PublishRequests` are present. The validity is checked using the time stamp of the `PublishRequests` as well as the `timeoutHint` stated in the `Request`. The resulting end time must be less than the current time of the server. If this is not the case, a `Response` with `OpcUa_BadTimeout` is sent back to the client. The `Request` is deleted in this case.
- ▶ **SubscriptionAcknowledgement that is included in the PublishRequest processes the OPC UA Server** if the `PublishRequest` is taken from the queue

REPUBLICSHREQUESTS

The **OPC UA Server** also supports the `RepublishRequests` in the event that the TCP connection between **OPC UA Client** and **OPC UA Server** is interrupted.

SYNCHRONOUS TIME

For `PublishRequests`, the OPC UA server takes the "`timeoutHint`" field into account.

If the OPC UA server has to send a `PublishResponse`, all existing `PublishRequests` in the queue (max. 10) are checked. This check validates whether the time stamp is more recent than the current time. If this is the case, the oldest `PublishRequest` from the queue is used.

If the queue contains `PublishRequests` whereby the time stamp in the `PublishRequest` + `timeoutHint` in the `PublishRequest` is the same or older than the current time in the OPC UA server, the `PublishRequests` are removed from the queue and for each `PublishRequest`, a `ServiceFault` with "`Bad_Timeout`" is sent.

It is therefore important that the system time of the OPC UA server and the system time of the OPC UA client is synchronized accordingly. For this time synchronization, it is recommended that, for both the OPC UA server and the OPC UA client, a common synchronization source is used. A common source can,

for example, be the same NTP server for server and client. Otherwise it is possible that the OPC UA server immediately rejects all `PublishRequests` from the OPC UA client and no value changes can be transferred.



Information

THE TIME STAMP NAME CONSISTS OF: PUBLISHREQUEST + TIMEOUTINT FROM THE PUBLISHREQUEST.

MONITORED ITEMS

The OPC UA server supports the adding, removal and modification of `MonitoredItems`.

If a queue of greater than 1 is requested from a OPC UA client, this request is supported. As a result, several value changes for the same variable are buffered between two publish intervals by the OPC UA server. A `PublishResponse` is used to send the value changes to the OPC UA client.

DATACHANGEFILTER - DEADBAND

The OPC UA server supports the `absolute deadband`.

READ REQUESTS

The OPC UA server also supports, in addition to communication by means of `Subscriptions`, `Read Requests`. In principle, communication by means of `Subscriptions` is preferred to communication by means of read requests. If an OPC UA client does not support subscriptions and sends read requests cyclically in order to get the current value, it is strongly recommended that the client uses the `RegisterNodes` call once for such nodes that are read cyclically.

This allows the OPC UA server to optimize communication with Runtime for cyclical `Read Requests`.

WRITE REQUESTS

The OPC UA server supports write requests. The set value limits of the variable are taken into account.

Attention: If a variable in the Energy Edition is linked to a command group and this variable is written by an OPC UA client, the command interlocking is not taken into account but the command variable is written directly.

13.9 Configuration file: INI entries for OPC UA

The configuration file `zenProcGateway.ini` contains this entries for OPC UA:

[OPCUA]

Entry	Description
SERVER_NAME=	Name of the OPC UA Server to which the Gateway establishes a connection.
SERVER_PORT=	Port for the communication to the OPC UA Server . Default: 4841
CERTIFICATES_PATH=	Path for the root folder with the certificate files and the respective subfolders. Default: %ProgramData%\PKI\CA\ Note: Entry must be deleted if a new certificate is to be created when Process Gateway is started.
SERVER_CERTIFICATE_LOCATION=	URI of the file with the public key of the server. The path is always stated as absolute. This public key is used by the client for the encryption of messages. Default: %ProgramData%\PKI\CA\certs\zenopcua.der Note: Entry must be deleted if a new certificate is to be created when Process Gateway is started.
SERVER_PRIVATE_KEY_LOCATION=	URI of the file with the private key of the server. The path is always stated as absolute. This private key is used by the server to decrypt the messages from the client. Default: %ProgramData%\PKI\CA\private\zenopcua.pem Note: Entry must be deleted if a new certificate is to be created when Process Gateway is started.
ONLY_SECURE_CONNECTIONS=	Selection of which type of connection (secure or not secure) is approved for establishing a connection. <ul style="list-style-type: none"> ▶ TRUE A connection to the server is only possible with a secure connection. ▶ FALSE A connection to the server is also possible with a non-secure connection. Default: TRUE
TRUST_ALL_CLIENT_CERTIFICATES=	Selection of which type of certificates (client or server certificates) is approved for establishing a connection. <ul style="list-style-type: none"> ▶ TRUE All client certificates are accepted and the

	<p>connection is made.</p> <ul style="list-style-type: none"> ▶ FALSE All clients that want to connect with the server must send a valid certificate. This certificate is also checked to see if it is in the list of known client certificates Certificate trust list path. <p>Default: TRUE</p>
USE_USER_AUTHENTICATION=	<p>Selection of whether the user administration of the current zenon project is to be used for identification when logging on to the server.</p> <ul style="list-style-type: none"> ▶ TRUE Clients can only log on to the server with a valid user ID and password. Note: To do this, the user administration must be activated and configured in zenon Editor. ▶ FALSE Anonymous login is possible. <p>Default: FALSE</p>
MAX_COUNT_OF_SUBSCRIPTIONS_PER_SESSION=	<p>Number of Subscriptions per session. Can be set between a minimum of 1 and a maximum of 100 Subscriptions.</p> <p>Default: 10</p>

[VARIABLES]

Entry	Description
PROJECT_x=	<p>Name of the project from which the variables come. x stands for the numbering within a project, starting with 0.</p> <p>Example: PROJECT_0=TEST</p>
PARAM_x=	<p>Parameters for project x. Numbering is from 0 upwards.</p> <p>Example: PARAM_0=80:INI.Driverforinternalvariables_ModemHwAdrSet</p>
COUNT=	<p>Number of configured variables.</p>
NAME_x=	<p>Name of the variable. x stands for the numbering of the project, starting with 0.</p> <p>Example: NAME_0=Driverforinternalvariables_ModemHwAdrSet</p>

2 = constants for namespace index

Is prescribed/issued by the system and must not be changed.

13.10 Error message

Error messages in the log file:

Error message	Debug Level	Description
Failed to validate client certificate	Error	<p>The client certificate could not be validated.</p> <p>It is either invalid (the date of validity may have been overwritten) or it was not found in the list of known client certificates.</p> <p>Additional information including an error number describes the error in more detail.</p>
Wrong user/password supplied while trying to activate session (User: %s)	Error	<p>User authentication failed.</p> <ul style="list-style-type: none"> ▸ %s: Name of the user to be authenticated.
Login of user '%s' succeeded.	Deep Debug	<p>User authentication successful.</p> <ul style="list-style-type: none"> ▸ %s: Name of the user to be authenticated.
Variable '%s' could not be added to OPC UA address space, because node '%s' with the same node identifier already exists	Error	<p>A variable defined by the user could not be created, because another variable with the same Identifier already exists.</p>
Failed to initialize server architecture: An internal error occurred as a result of a programming or configuration error. (0x80020000)	Error	<p>The file format for one or more of the following files is not present in a valid format:</p> <ul style="list-style-type: none"> ▸ Certificate ▸ Private Key ▸ Certificate Revocation List

%1 PublishResponse with OpcUa_BadNoSubscriptions sent because no subscriptions are created in session %2	Error	<p>There is no valid Subscription for a publish request or the session was deleted. The OPC UA server sends a messages to the OPC UA client for all active PublishRequests if the last subscription was deleted by the server or the client.</p> <ul style="list-style-type: none"> ▶ %1 Number of PublishResponses that have been sent to a client with the result value OpcUa_BadNoSubscription. ▶ %2 Name of the session from which the PublishResponse has been sent.
%1 PublishResponse with OpcUa_BadTimeout sent in session %2	Error	<p>The timestamp + timeoutHint in the PublishRequest that is provided by the client is older than the current time on the OPC UA server. Possible causes: the timeoutHint in the PublishRequest is too short, the OPC UA client provides more PublishRequests than are needed and the PublishRequests expire or the time of the computer with the OPC UA server and the time of the computer with the OPC UA client are not synchronous.</p> <ul style="list-style-type: none"> ▶ %1 Number of PublishResponses that have been sent to a client with the result value OpcUa_BadTimeout. ▶ %2 Name of the session from which the PublishResponse has been sent. <p>Note: The PublishRequest is deleted.</p>
%1 advise/unadvise thread procedure	DEBUG	<p>Start or stop of an Advise or Unadvise thread:</p> <ul style="list-style-type: none"> ▶ %1 Entering Thread started ▶ %1 Leaving Thread stopped
%1 shutdown	DEBUG	<p>Start and end of a shutdown sequence:</p> <ul style="list-style-type: none"> ▶ %1 beginning Shutdown start ▶ %1 finished Shutdown ended
%1 %2 request	DEBUG	<p>Start or end of a Request when reloading Runtime:</p> <ul style="list-style-type: none"> ▶ %1 Processing

		<p>Start of a request</p> <ul style="list-style-type: none"> ▶ %1 Finished <p>End of a request</p> <ul style="list-style-type: none"> ▶ %2 ACTIVE <p>Request is active.</p> <ul style="list-style-type: none"> ▶ % 2 INACTIVE <p>Request is inactive.</p>
Server shut down.	DEBUG	Notification that the server has been fully shut down.
The runtime does not provide archive variable [%1] of project [%2]	ERROR	<p>The archive of a configured archive variable can no longer be reached in zenon Runtime.</p> <ul style="list-style-type: none"> ▶ %1 Name of the archive variable ▶ %2 Project name
Reading configuration from %1	MSG	<p>The configuration is read from the INI file.</p> <ul style="list-style-type: none"> ▶ %1 Name of the INI file.
Linking configuration to the Runtime	MSG	The variables configured in Process Gateway are linked to zenon Runtime.
some configured variables are not linked to the Runtime	ERROR	When linking, it is established that one or more configured variables are not available in zenon Runtime. The projects or variables that are not available are listed.
Project [%1] is not linked to the Runtime	MSG	<p>The project configured with the variables is not available in zenon Runtime.</p> <ul style="list-style-type: none"> ▶ %1 Project name <p>Possible causes: The project is not loaded or has been deleted.</p>
Variable [%1] of Project [%2] is not linked to the Runtime	MSG	<p>The variable is not available in zenon Runtime:</p> <ul style="list-style-type: none"> ▶ 1% Variable name ▶ %2 Project name
Could not write value [%1] to key [%2] in section [%3]	ERROR	<p>Error when writing a value.</p> <ul style="list-style-type: none"> ▶ %1

		<p>Value that is to be written.</p> <ul style="list-style-type: none"> ▶ %2 Key in which the value is to be written. ▶ %3 Section of the key in which the value is to be written.
No separator for archive flag [,] found in value [%s] of key [%s]	ERROR	<p>Missing separator character for a value. The variable concerned can therefore not be applied in the configuration:</p> <ul style="list-style-type: none"> ▶ %1 Variable name ▶ %2 Key of the value

13.11 Status mapping

In certain situations, the OPC UA server communicates a different `StatusCode` for the value (Value Attribute). This concerns each `node` that represents the variable.

The table below lists the zenon status with the corresponding **OPC UA StatusCode**.

Status of the variable in the Runtime	StatusCode Value Attribute OPC UA Node
INVALID	0x80050000 BadCommunicationError
ALT_VAL	0x40910000 UncertainSubstituteValue
OFF	0x808C0000 BadSensorFailure
OV_870	0x803C0000 BadOutOfRange
N_UPDATE	0x80310000 BadNoCommunication

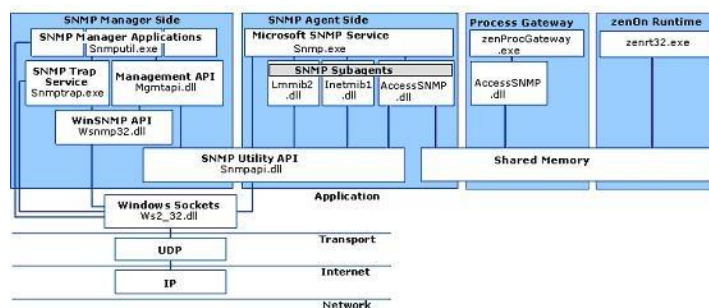
14. SNMP agent

14.1 SNMP architecture

The Windows SNMP agent supports SNMP version 2 (SNMPv2c). It also supports the SNMP functions (GET, GETNEXT, GETBULK, SET, TRAP).

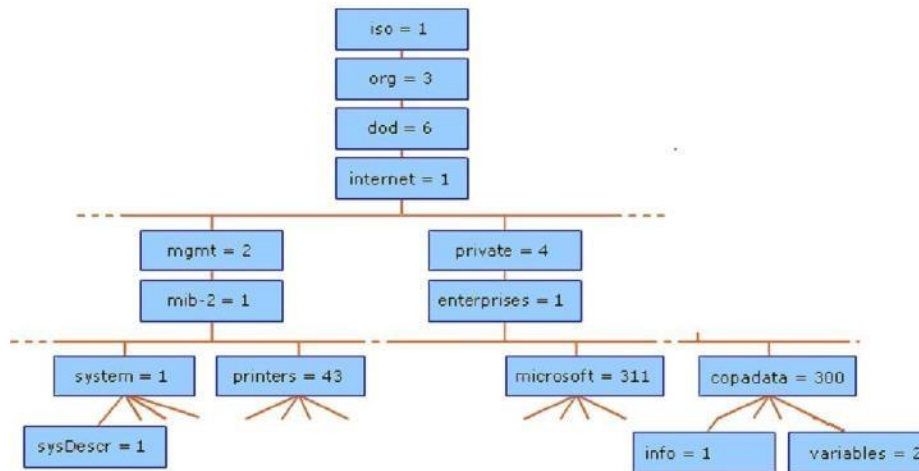
More details about Microsoft's implementation of the SNMP can be found on the Microsoft TechNet Website: "How SNMP Works" [https://technet.microsoft.com/en-us/library/cc783142\(v=ws.10\).aspx](https://technet.microsoft.com/en-us/library/cc783142(v=ws.10).aspx) ([https://technet.microsoft.com/en-us/library/cc783142\(v=ws.10\).aspx](https://technet.microsoft.com/en-us/library/cc783142(v=ws.10).aspx))

WINDOWS OPERATING SYSTEM COMPONENTS WITH A CONNECTION TO THE PROCESS GATEWAY AND THE RUNTIME:



14.2 Management Information Base (MIB)

The Management Information Base (MIB) is organized in the form of a tree structure, in which the SNMP agent provides the variables = (OID – Object Identifier). Below is the MIB structure, with additional variables visible, which have been provided by the Process Gateway.



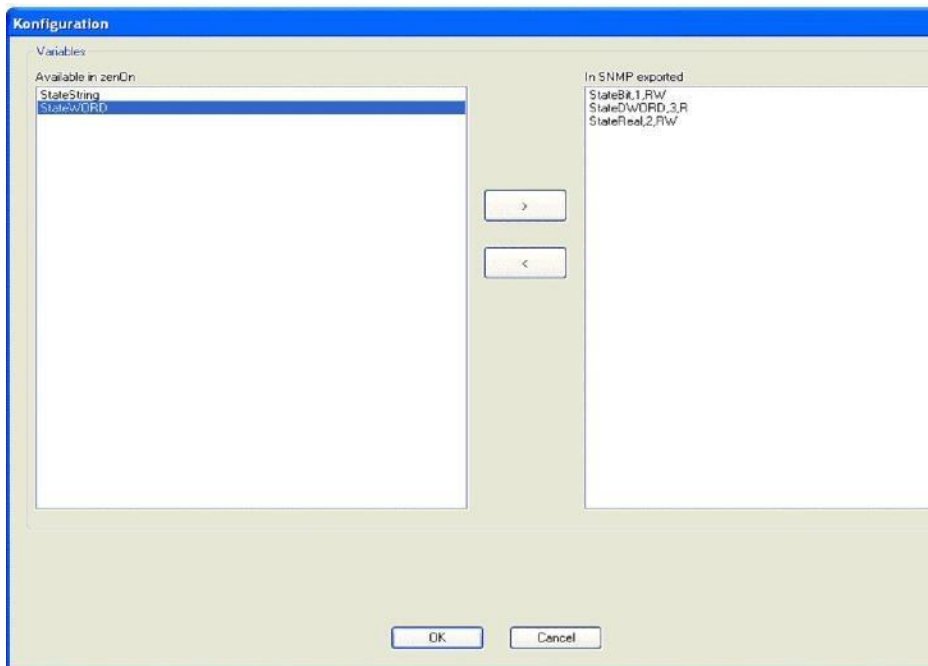
14.3 MIB structure

Info block structure is always present				
OID	Description	Data type	Access	Change is sent as Trap
1.3.6.1.4.1.300.1.1.0	copadata.info.statusinfo, Runtime is (Online, Offline)	string	read	yes
1.3.6.1.4.1.300.1.2.0	copadata.info.statusval, Runtime is 1=Online 0=Offline	integer	read	yes
1.3.6.1.4.1.300.1.3.0	copadata.info.watchdog	integer	read	no
1.3.6.1.4.1.300.1.4.0	copadata.info.project, name of project which the Process Gateway reads variables from	string	read	no

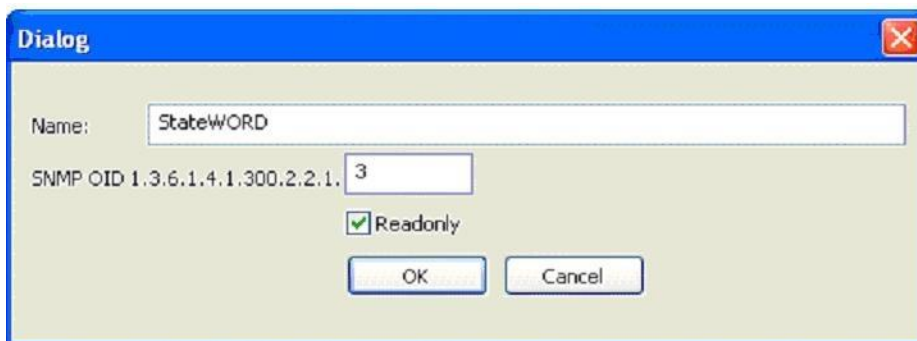
Variable structure (table) is dynamically adjusted, according to the configuration of the Process Gateway				
OID	Description	Data type	Access	Change is sent as Trap
1.3.6.1.4.1.300.2.1.0	copadata.variables.ifNumber, number of configured variables=OID	integer	read	yes
1.3.6.1.4.1.300.2.2.1.X	copadata.variables.ifTables.ifIndex, configured OID index from the Process Gateway configuration	integer	read	no
1.3.6.1.4.1.300.2.2.2.X	copadata.variables.ifTables.ifName, variable name	string	read	no
1.3.6.1.4.1.300.2.2.3.X	copadata.variables.ifTables.ifValue, variable value as Integer. All numerical data types are formatted to an SNMP Integer data type. ATTENTION: the decimal places of REAL and DOUBLE are lost.	integer	according to configuration in Process Gateway	no
1.3.6.1.4.1.300.2.2.4.X	copadata.variables.ifTables.ifValueAs String, variable value as String. All numerical data types are formatted to an SNMP String data type.	string	according to configuration in Process Gateway	yes
1.3.6.1.4.1.300.2.2.5.X	copadata.variables.ifTables.ifStatus, variable status value	integer	read	yes
1.3.6.1.4.1.300.2.2.6.X	copadata.variables.ifTables.ifTimestamp, variable timestamp	integer	read	yes

14.4 Module configuration

The following dialog appears after the **Settings** button is clicked:



Here the variables that should be connected can be selected. On the left-hand side all variables which are available in zenon are displayed. You can select them there and move them to the SNMP export list with the button >. With the button < they can be removed again. If you move a variable to the export list, a dialog for the name assignment of the zenon variable to an OID address in the SNMP structure appears:



The variables selected in this way are then available on the SNMP server and are continuously synchronized between zenon and the SNMP OID structure. For every OID address you can define whether it can be write-accessed via SNMP SET. The exact structure of every OID address is described in the chapter "MIB structure".

14.5 Configuration file: specific entries for Access SNMP

The configuration file must be in the system folder.

Parameters	Description
[SNMP]	
REFRESH RATE	Update time in milliseconds

Parameters	Description
[VARIABLES]	
COUNT	number of variables to be exported
NAME_n	name of the variable with the number n ($0 \leq n \leq \text{COUNT}$)
PARAM_n	Parameter SNMP OID table offset and the information R=read only / RW=read and write with the number n ($0 \leq n < \text{COUNT}$)

EXPORT OF VARIABLES

For the successful export the project name in `zenProcgateway.ini` in folder `%CD_SYSTEM%` must be entered correctly. Enter it here:

```
[DEFAULT]
PROJECT=
```

Attention: If the entry is wrong or missing the gateway is closed without any error messages. Thus the variable is not available.

14.6 Configuration of the Windows SNMP agent

To make sure that the Windows SNMP Agent (**SNMP.exe**) provides the zenon MIB partial structure, the Process Gateway file **AccessSNMP.dll** in the registry must be configured accordingly:

1. Installation of the Windows SNMP service via the control panel:
 - Via module: Programs and functions - >Activate or deactivate Windows functions.

Note: With Windows 8, the computer must be restarted once the SNMP service has been added.
2. Installation of Process Gateway with the file `AccessSNMP.dll` in the folder
`%Program Files (x86)%\COPA-DATA\zenon 8.00`
3. Add the following keys and character strings using the registry editor. The paths and content are sometimes different for 32-bit systems and 64-bit systems:

32-Bit:

- **[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\SNMP\Parameters\ExtensionAgents]**
Name=ProcessGateway
Wert=SOFTWARE\COPA-DATA\SNMP\CurrentVersion
- **[HKEY_LOCAL_MACHINE\SOFTWARE\COPA-DATA\SNMP\CurrentVersion]**
Name=Pathname
Wert=C:\Program Files\COPA-DATA\zenon 8.00 SP0\AccessSNMP.dll

64-Bit:

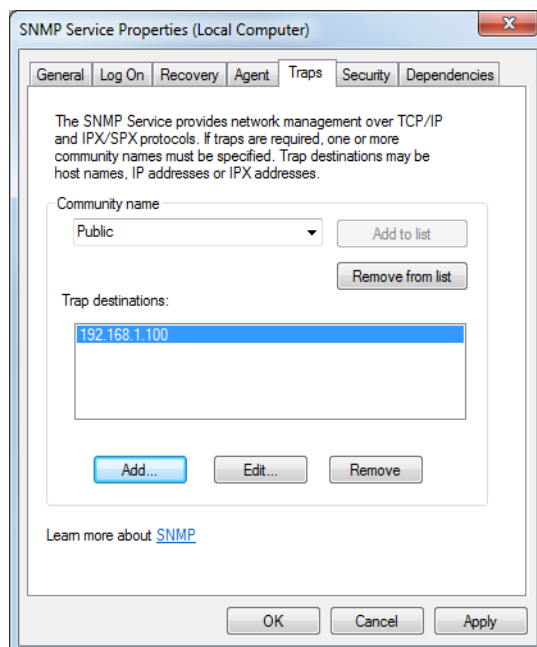
- **[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\SNMP\Parameters\ExtensionAgents]**
Name=ProcessGateway
Wert=SOFTWARE\COPA-DATA\SNMP\CurrentVersion
- **[HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\COPA-DATA\SNMP\CurrentVersion]**
Name=Pathname
Wert=C:\Program Files (x86)\COPA-DATA\zenon 8.00 SP0\AccessSNMP.dll

4. Start the SNMP service.

This loads the process gateway file AccessSNMP.dll via the registry configuration.

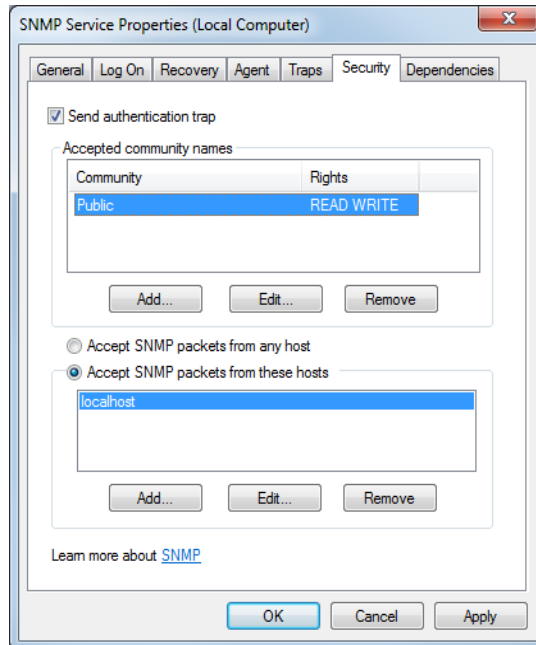
Further configuration settings for the Windows SNMP service: Control panel -> Administration-> Services-> SNMP service-> Properties

- a) **Traps:** tab Defines the IP address in the network to which traps are sent:



- **Communityname:** Must be identical on manager and agent
- **Trap destinations:** Addresses of the destinations as host names, IP addresses or IPX addresses

a) **Security:** tab Defines the rights with which an SNMP client can access the SNMP agent:



- **Rights:** Must be set to "READ WRITE"
- **Community:** Must be identical on manager and agent

15.SQL

15.1 Module configuration

The configuration of the database connection consists of two parts.

1. OLE-DB connection: This OLE-DB connection can be defined in the configuration dialog using the selection button for the connect string. Then the standard dialog for data connection properties is opened.
2. Additionally the name of the table can be defined, if the standard name ONLINE_VALUES should not be used.

This can be set with the **TABLE** entry in the **[DATABASE]** section of the configuration file (`zenProcGateway.ini`). The settings are taken over by the add-on when starting it.



Attention

If table names are defined manually in the configuration file (zenProcGateway.ini), these table names must also be amended or created in the database by hand.

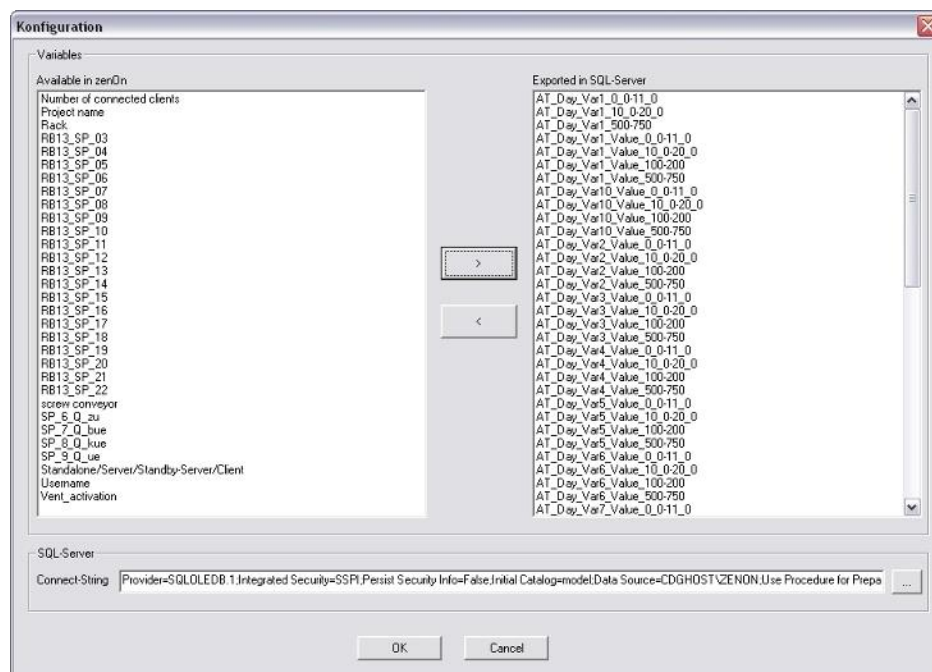
ORACLE DATABASES

Use of the Process Gateway and `AccessSQL.dll` in conjunction with Oracle databases can cause errors: If it is possible, with some ODBC drivers, that no online values are written to the database. The advanced ODBC logging then issues the error message "**1843**".

The problem can be solved by changing the ODBC driver. Example: ODBC driver 9.02.00.05 for Oracle 9.2.

15.2 Variables selection

In the configuration dialog the zenon variables that should be written to the SQL process image can be defined. In order to do this the variables available in the zenon project - listed in the left part - have to be added to the image - listed in the right part.



15.3 Creating tables

After closing the configuration dialog with **OK**, the table structure of the SQL database is automatically generated. For the case, that it is created by hand:

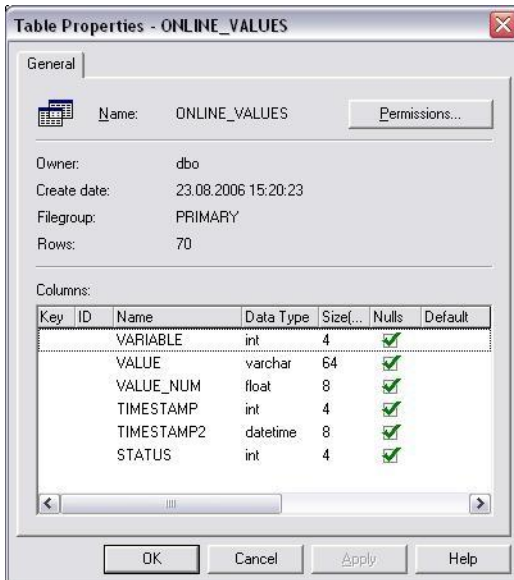


Table Properties - ONLINE_VALUES

General

Name: ONLINE_VALUES Permissions...

Owner: dbo

Create date: 23.08.2006 15:20:23

Filegroup: PRIMARY

Rows: 70

Columns:

Key	ID	Name	Data Type	Size...	Nulls	Default
		VARIABLE	int	4	<input checked="" type="checkbox"/>	
		VALUE	varchar	64	<input checked="" type="checkbox"/>	
		VALUE_NUM	float	8	<input checked="" type="checkbox"/>	
		TIMESTAMP	int	4	<input checked="" type="checkbox"/>	
		TIMESTAMP2	datetime	8	<input checked="" type="checkbox"/>	
		STATUS	int	4	<input checked="" type="checkbox"/>	

OK Cancel Apply Help

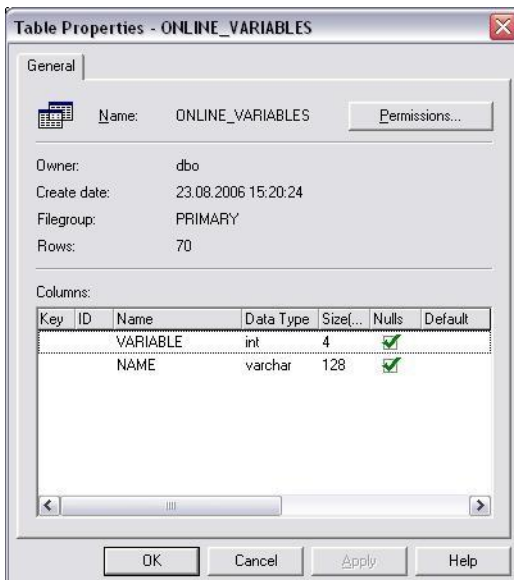


Table Properties - ONLINE_VARIABLES

General

Name: ONLINE_VARIABLES Permissions...

Owner: dbo

Create date: 23.08.2006 15:20:24

Filegroup: PRIMARY

Rows: 70

Columns:

Key	ID	Name	Data Type	Size...	Nulls	Default
		VARIABLE	int	4	<input checked="" type="checkbox"/>	
		NAME	varchar	128	<input checked="" type="checkbox"/>	

OK Cancel Apply Help

15.4 Operation in a redundant system

In redundant operation two alternatives are available. In the first variation the SQL database runs locally on both servers and the Process Gateway also runs on both servers. As only online values are handled

no further alignment between the databases is necessary. The big disadvantage of this version is the high network load that is a result from the additional reading of online values on the Standby Server.

In the second version the Process Gateway only runs on the Runtime server and writes data to a local or a remote SQL database. In this alternative no additional network load between the Runtime Server and the Standby Server occurs and in the case of the remote SQL database the Server is not strained by external requests to that database.

15.5 Configuration file: specific entries for AccessSQL

The configuration file must be in the system folder.

Parameter	Description
[DATABASE]	
INITSTRING	OLE-DB connect-string to the SQL database
TABLE_ONL	name of the table that receives the process image
TABLE_VAR	name of the table that receives the variable IDs
REFRESHRATE	maximum refresh rate in milliseconds

Parameter	Description
[VARIABLES]	
COUNT	number of variables to be exported
NAME_n	Name of variable number n. Rules: <ul style="list-style-type: none"> ▶ starts with 0 ▶ Must be consecutive without interruption

15.6 Table format

15.6.1 Table for process image

VARIABLE	int[4]	ID of the zenon variable
VALUE	varchar[64]	current value as string
VALUE_NUM	float	current value as float
TIMESTAMP	int[4]	time stamp of the current value (as UNIX time)
TIMESTAMP 2	datetime	time stamp of the current value (as datetime)
STATUS	int[4]	status word of the current value

15.6.2 Table for variable IDs

VARIABLE	int[4]	ID of the zenon variable
NAME	varchar[128]	Name of the zenon variable.