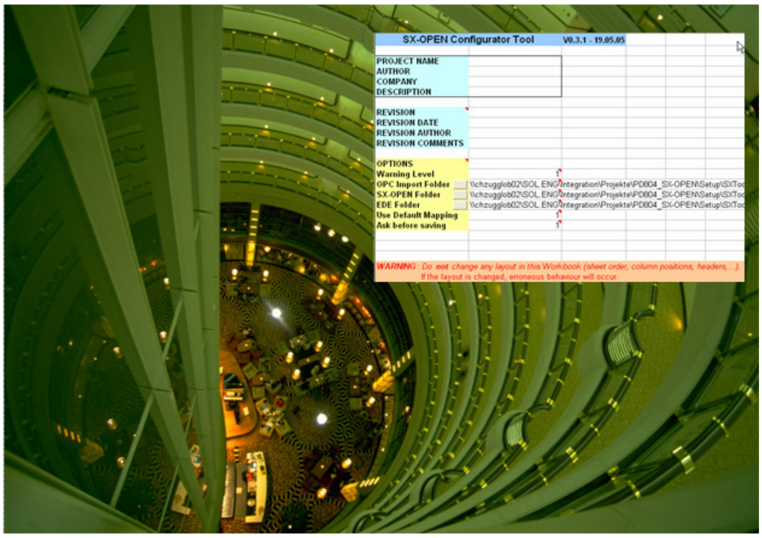
SIEMENS



Desigo™ SX Open
SX Configurator, V3
User's Guide

CM110702en_09
2015-10-21

Siemens Building Technologies

Table of contents

1 About this document

SX Configurator is supplied as an integral component of the SX Open software package, and is a necessary tool for the configuration of this software (refer also to section 2).

This manual describes the use of SX Configurator. It describes the step-by-step process for configuring SX Open (section 3) and contains detailed background information on importing and compiling OPC data, and on creating the various worksheets in the SX Configurator file (sections 4 to 5.14).

Validity

This document is valid for Version V2.1 of SX Configurator.
(Refer to the text file "**Readme.txt** for other important information.)

Target group

The document is written for staff at HQ and in the Regional Companies concerned with the use and configuration of SX Open, in particular:

- Design engineers and engineering specialists
- Commissioning and service engineers
- Product managers and those responsible for supporting the products
- Developers at HQ

Assumed knowledge

It is assumed that the users of SX Configurator already have an adequate understanding of the following specialized fields:

- HVAC and building automation and control in general
- Network and communications technology (Ethernet/LAN network)
- OPC
- BACnet
- The Desigo system
- Microsoft Office and, specifically, Microsoft Excel

Purpose

The objective of this manual is to help users to configure SX Open and to understand and implement the associated optional features.

1.1 Revision history

Version	Date	Changes	Section
3	**2015-10-21**	• **Cyber security disclaimer** • **Excel macros** • **Upload of properties**	**1.3** **3.1** **5.14**
3	2014-06-23	• Changes for V3 • Various corrections	Throughout
2.1	2015-10-21	New product name SX Open	Throughout
		Correcting Typos, added clarifications	Throughout
		Change default notification class to 31	3.7.6
		New sections schedule object	3.7.8, 5.2, 5.2.8
		Deleted figure 3.1	3.10
		Clarification of alarm-values	5.2.6
		Hint for more than one SX Trend Server	5.2.7
		No alarm suppression for “trended” objects	5.2.9
		Removed restriction to read-only mapping for VT_BSTR items	5.2.10
		Detailed description of BACnet client sheet	5.12
		Calculation of Hash-Values done by SX Configurator	5.13
		New chapter for example configurations	7
		New FAQ entry, default BACnet object mapping	8

1.2 Before you start

Trademarks

The trademarks used in this document are listed together with their legal owners in the following table. The use of these trademarks is subject to international and national statutory provisions.

Trademarks	Legal owner
BACnet™	American National Standard (ANSI/ASHRAE 135-1995)
Microsoft … Office Excel	Microsoft Corporation see http://www.microsoft.com/TRADEMARKS/t-mark/nopermit.htm
Windows …	Microsoft Corporation
OPC	OPC Foundation, Scottsdale AZ, USA

All the product names listed are trademarks (™) or registered trademarks (®) of their respective owners, as listed in the table. Further to the notes in this section, and to facilitate the reading of the text, these trademarks will not be indicated elsewhere in the text (e.g. by use of symbols such as ® or ™).

Copyright

Quality assurance

These documents were prepared with great care.

- The contents of all documents are checked at regular intervals.
- Any corrections necessary are included in subsequent versions.
- Documents are automatically amended as a consequence of modifications and corrections to the products described.

Please make sure that you are aware of the latest document revision date.

If you find lack of clarity while using this document, or if you have any criticisms or suggestions, please contact the product manager in your nearest branch office. The addresses of the Siemens regional companies are available at www.siemens.com.

Document use / request to the reader

Before using our products, it is important that you read the documents supplied with or ordered at the same time as the products (equipment, applications, tools etc.) carefully and in full.

We assume that persons using our products and documents are authorized and trained appropriately and have the technical knowledge required to use our products as intended.

More information on the products and applications is available:

- On the intranet (Siemens employees only) at https://workspace.sbt.siemens.com/content/00001123/default.aspx
- From the Siemens branch office near you www.siemens.com or from your system supplier
- From the support team at headquarters fieldsupport-zug.ch.sbt@siemens.com if there is no local point of contact

Siemens assumes no liability to the extent allowed under the law for any losses resulting from a failure to comply with the aforementioned points or for the improper compliance of the same.

1.3 Cyber security disclaimer

Siemens products and solutions provide IT-specific security functions to ensure the secure operation of building comfort, fire safety, security management and physical security systems. The security functions on these products and solutions are important components of a comprehensive security concept.

It is, however, necessary to implement and maintain a comprehensive, state-of-the-art security concept that is customized to individual security needs. Such a security concept may result in additional site-specific preventive action to ensure that the building comfort, fire safety, security management or physical security system for your site are operated in a secure manner. These measures may include, but are not limited to, separating networks, physically protecting system components, user awareness programs, in-depth security, etc.

For additional information on building technology security and our offerings, contact your Siemens sales or project department. We strongly recommend signing up for our security advisories, which provide information on the latest security threats, patches and other mitigation measures.

http://www.siemens.com/innovation/de/technologiefokus/siemens-cert/cert-security-advisories.htm

1.4 Printing conventions

1.4.1 Printing conventions for the BACnet terms

BACnet objects

BACnet objects (standard and proprietary) are not written or marked out in any special way in the manual.

Example:

Notification class

BACnet properties

The first time it occurs in a section of this manual, a BACnet property is written in full, followed by the abbreviated form in square brackets. After this, the property is referred to only by its abbreviated name or "identifier".

Example: Present_Value [PrVal]

1.5 Related documents

The following other documents are available for Desigo SX Open:

Reference	Title	Document No.
[1]	SX Open Engineering Guideline	CM110700
[2]	Excel project tool DIEPT	CM110634
[3]	DESIGO INSIGHT-OPEN OPC Import Tool User's guide	CA2Y9751
[4]	DESIGO INSIGHT-OPEN OPC Import Tool	CA2N9751
[5]	DESIGO Technical principles, Section 7 Alarm concept	CM110664

2 SX Open – Integration into BACnet

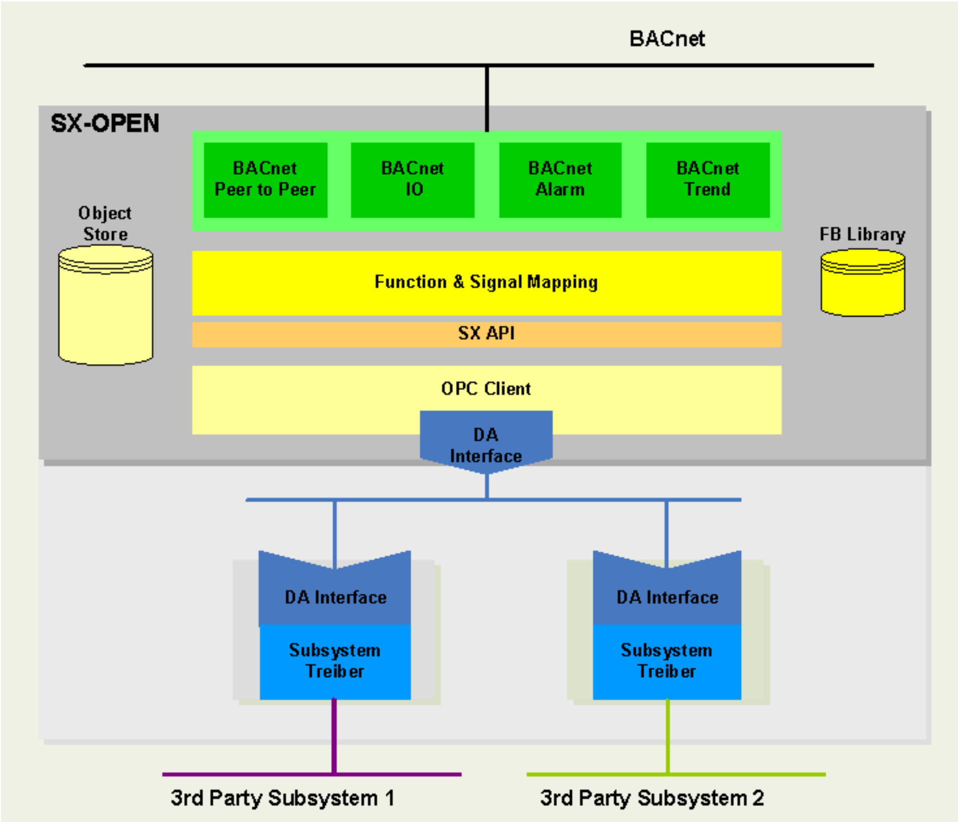


Figure 2-1

SX Open is a software package in the Desigo Open family, and is used to integrate third-party subsystems or devices with an OPC interface into a BACnet system.

Integration into Desigo

SX Open enables devices or complete systems from third-party manufacturers to be integrated into the Desigo system via the OPC data access interface.

SX Open is a configurable gateway between OPC and BACnet, which provides access to the integrated third-party data points both for the Desigo PX automation stations and for the Desigo Insight management station. For further details, refer to the Engineering Guideline, [1].

SX Open configuration

SX Open requires information about the OPC data to be integrated, and needs to know how this data is to be represented on the BACnet side. This is why SX Open has to be configured.
In addition to these mapping rules, other optional functions can also be enabled.
The tool **SX Configurator** was developed specifically for this configuration process.

SX Configurator

SX Configurator is a Microsoft Excel file (**SX-Configurator.xls**) with a fixed, predefined structure.

The file contains worksheets and functions designed to support the engineering workflow.

- Import of OPC data descriptions
- Compilation (translation of the configuration into a form readable by SX)
- Export of BACnet object descriptions for XWP and Desigo Insight
- Creation of the downloadable configuration for SX Open.

This file is part of the SX Open package, and is copied into the subfolder ...**Config** of SX Open datapath when SX Open is set up.

As a rule, SX Configurator also runs from this subfolder. However, the configuration process can also be carried out without the SX Open setup. It is sufficient simply to copy the file **SX-Configurator.xls**.

Note

Note that in this case, after you have configured SX Open, the resulting *.CSV files must be copied in CSV format (CSV = comma separated values) to the subfolder **…\Config**. When SX Open is started, it looks for the configuration in this location.

However, we recommend that you carry out the engineering on the same PC on which SX Open is installed.

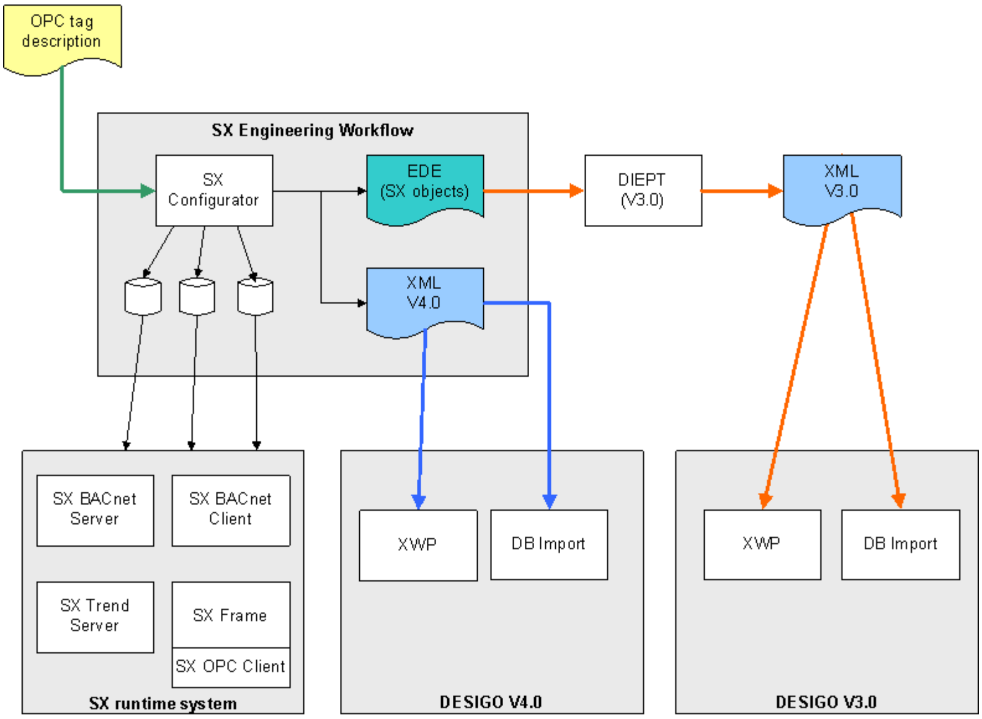


Figure 2-2

3 SX configuration workflow

The workflow consists of the following main activities:

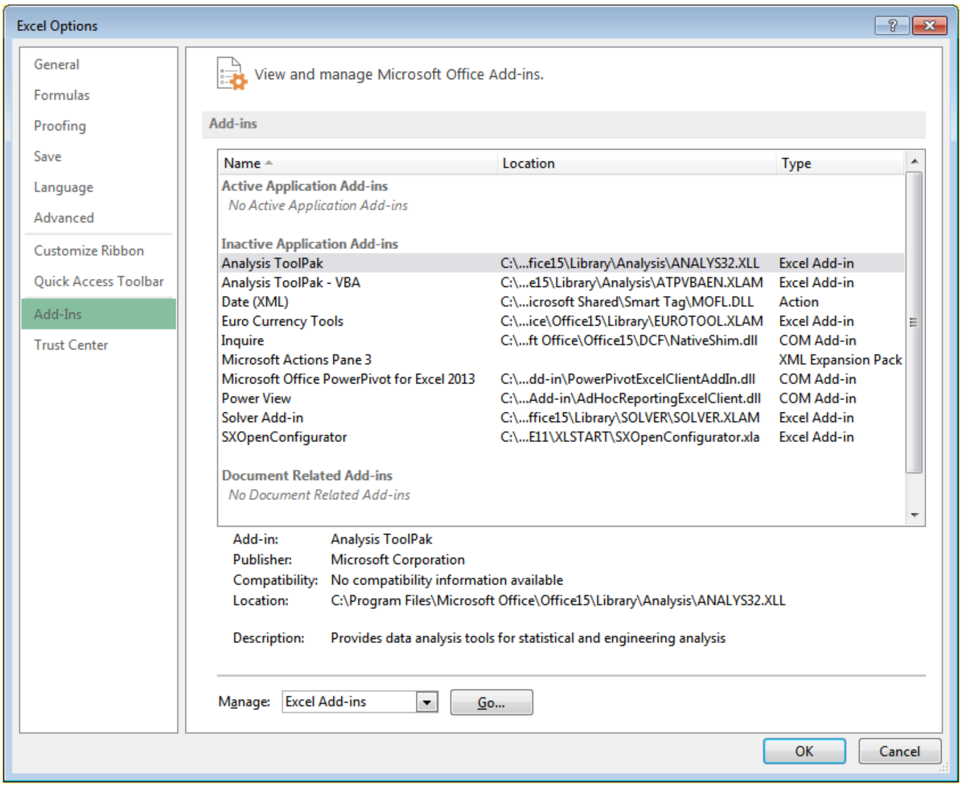
- Opening SX Configurator (Section 3.1)
- Checking global Excel settings (Section 3.2)
- Cover sheet (Section 3.3)
- Importing data (Section 3.4 to 3.6)
- Editing the main configuration sheet (Section 3.7)
- Editing other worksheets (refer to sections 3.8 and 3.11)

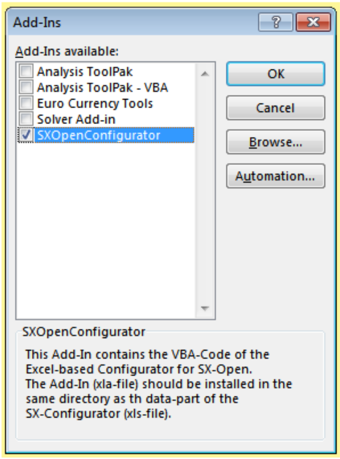
3.1 Opening SX Configurator

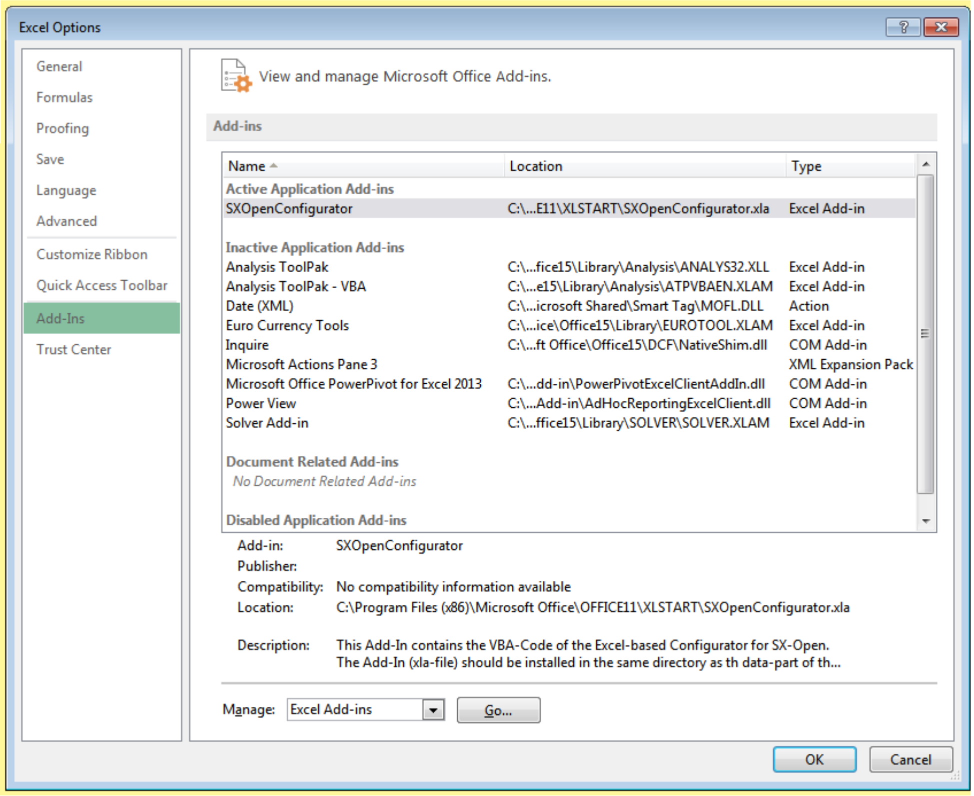
Enable Excel macro

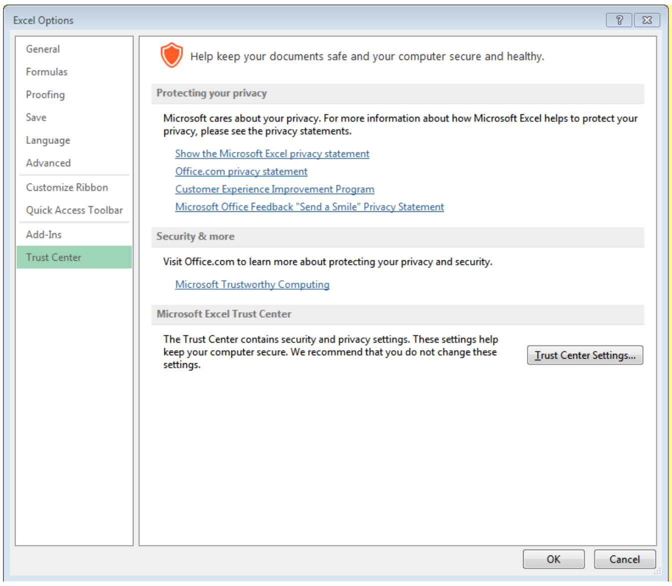
To have a running macro (Excel 2010 and Excel 2013 64 bit)

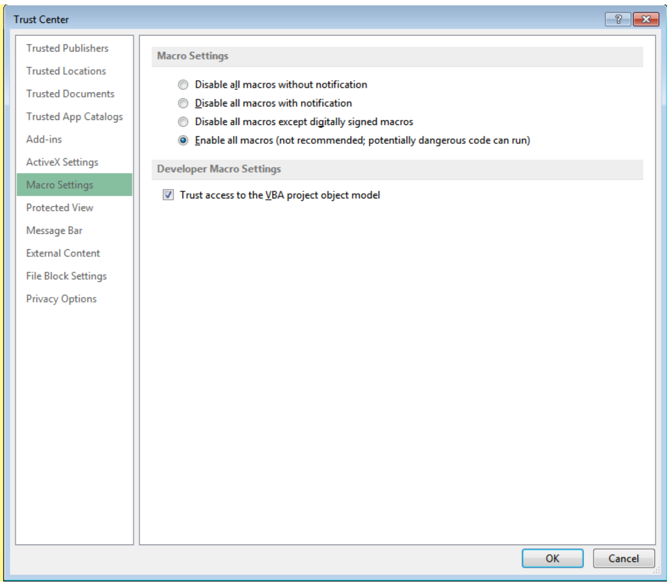
- Enable the Macro in the Excel Options > Trust Center > Macro Settings
- Activate the Add-in in the Excel Options > Add-In > Press "Go" button > Activate the SxOpenconfigurator Add-in in the list





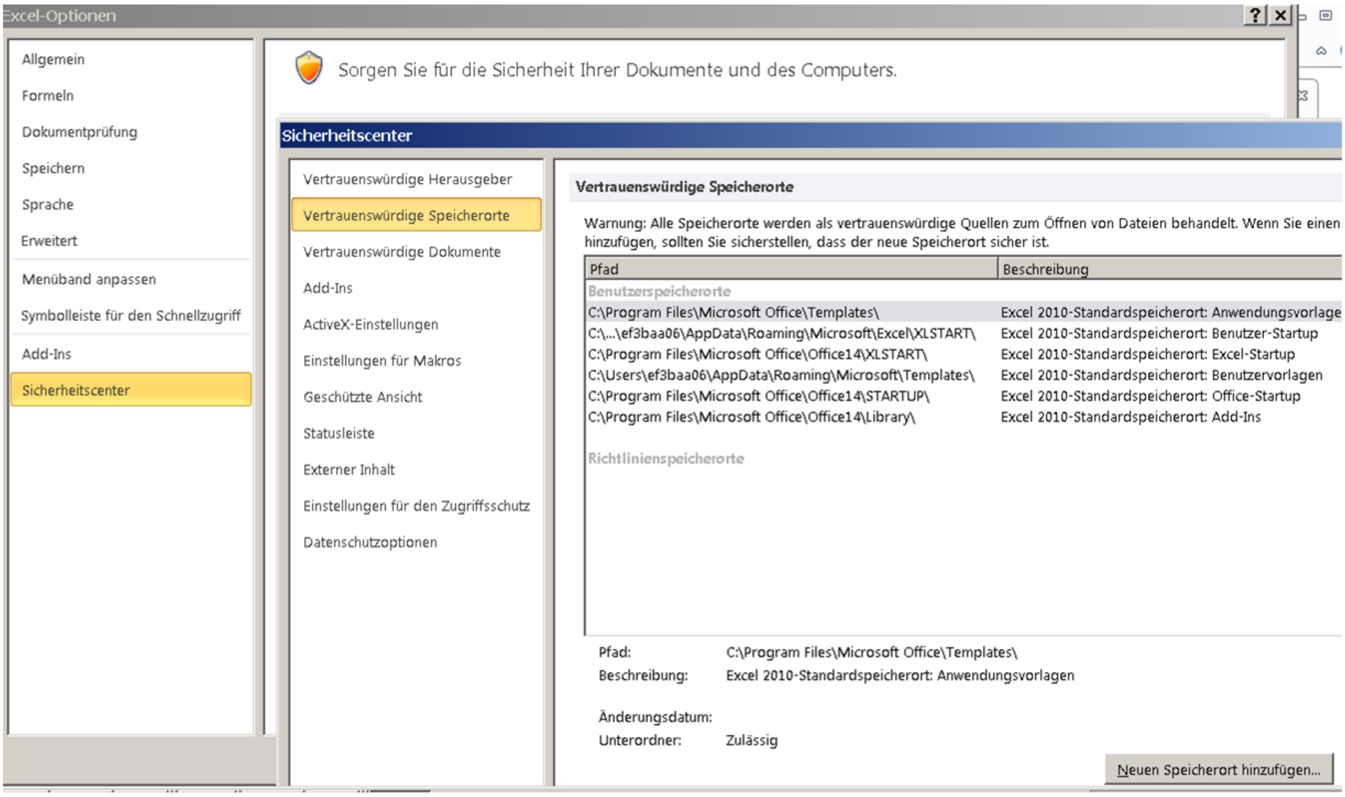

Excel Options
General
Formulas
Proofing
Save
Language
Advanced
Customize Ribbon
Quick Access Toolbar
Add-Ins
Trust Center
View and manage Microsoft Office Add-ins.
Add-ins
Name
Location
Type
Active Application Add-ins
SXOpenConfigurator C:\...E11\XLSTART\SXOpenConfigurator.xla Excel Add-in
Inactive Application Add-ins
Analysis ToolPak C:\...fice15\Library\Analysis\ANALYS32.XLL Excel Add-in
Analysis ToolPak - VBA C:\...e15\Library\Analysis\ATPVBAEN.XLAM Excel Add-in
Date (XML) C:\...icrosoft Shared\Smart Tag\MOFL.DLL Action
Euro Currency Tools C:\...ice\Office15\Library\EUROTOOL.XLAM Excel Add-in
Inquire C:\...ft Office\Office15\DCF\NativeShim.dll COM Add-in
Microsoft Actions Pane 3 XML Expansion Pack
Microsoft Office PowerPivot for Excel 2013 C:\...dd-in\PowerPivotExcelClientAddIn.dll COM Add-in
Power View C:\...Add-in\AdHocReportingExcelClient.dll COM Add-in
Solver Add-in C:\...ffice15\Library\SOLVER\SOLVER.XLAM Excel Add-in
Document Related Add-ins
No Document Related Add-ins
Disabled Application Add-ins
Add-in: SXOpenConfigurator
Publisher:
Compatibility: No compatibility information available
Location: C:\Program Files (x86)\Microsoft Office\OFFICE11\XLSTART\SXOpenConfigurator.xla
Description: This Add-In contains the VBA-Code of the Excel-based Configurator for SX-Open.
The Add-In (xla-file) should be installed in the same directory as th data-part of th...
Manage: Excel Add-ins
Go...
OK
Cancel


Excel Options
General
Formulas
Proofing
Save
Language
Advanced
Customize Ribbon
Quick Access Toolbar
Add-Ins
Trust Center
Help keep your documents safe and your computer secure and healthy.
Protecting your privacy
Microsoft cares about your privacy. For more information about how Microsoft Excel helps to protect your privacy, please see the privacy statements.
Show the Microsoft Excel privacy statement
Office.com privacy statement
Customer Experience Improvement Program
Microsoft Office Feedback "Send a Smile" Privacy Statement
Security & more
Visit Office.com to learn more about protecting your privacy and security.
Microsoft Trustworthy Computing
Microsoft Excel Trust Center
The Trust Center contains security and privacy settings. These settings help keep your computer secure. We recommend that you do not change these settings.
Trust Center Settings...
OK
Cancel


Trust Center
Trusted Publishers
Trusted Locations
Trusted Documents
Trusted App Catalogs
Add-ins
ActiveX Settings
Macro Settings
Protected View
Message Bar
External Content
File Block Settings
Privacy Options
Macro Settings
Disable all macros without notification
Disable all macros with notification
Disable all macros except digitally signed macros
Enable all macros (not recommended; potentially dangerous code can run)
Developer Macro Settings
Trust access to the VBA project object model
OK
Cancel

Before opening SX-Configurator.xls in Excel 2010 / 2007, make sure to add the folder where the configurations are, to the "Trustworthy locations".
Excel options > Security center > Security center settings > press button "Add new location".

Otherwise the button "Add new location" must be pressed each time Excel 2007 / 2010 is started in order to view the SX configuration toolbar.



Open SX-Configurator.xls

To create a new SX configuration, open the file **SX-Configurator.xls**. This file serves a template, and must not be overwritten.

When you open the file **SX-Configurator.xls**, five new buttons automatically appear on the toolbar.

Get OPC data

Open the dialog box for importing OPC data.

Create SX-OPEN files

Click to save the SX Open configuration and compile (translate) it into an SX-readable form.

Upload properties

Import values from current installation to preserve changes done by customer.

Cancel current operation

Cancel the current operation.

Note

If you cancel the import of OPC data, you must delete any data already imported, in order to maintain the (internal) consistency of the tool. A dialog box is displayed indicating that data will be lost. The data is not deleted until this warning is acknowledged.

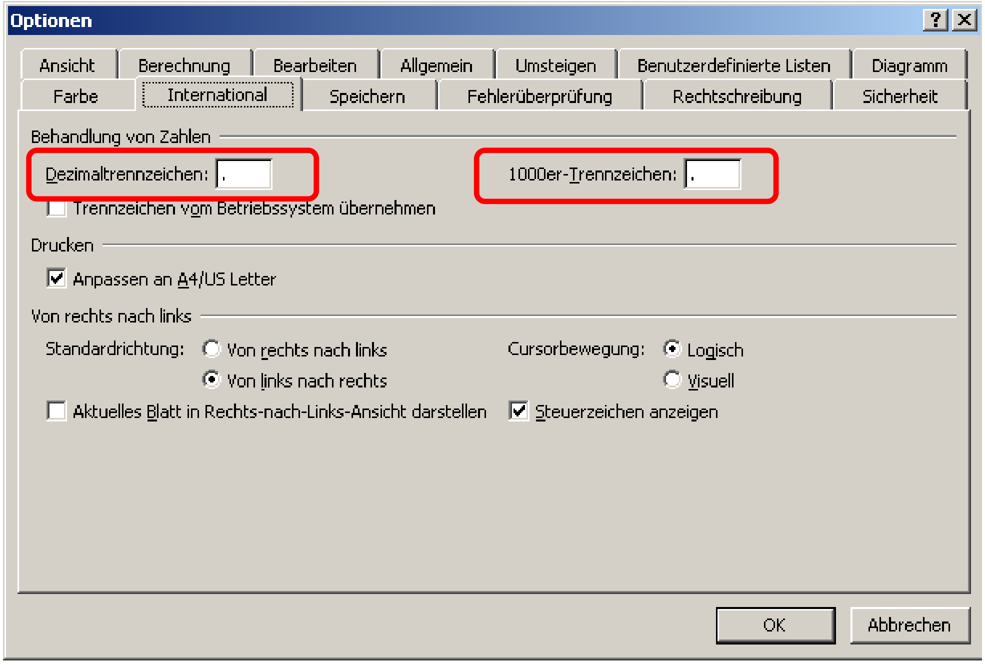
Copying data

If the SX configuration is to be carried out on another PC, copy the complete folder **…\config**.

If the SX Configurator file is used on another computer, the fields **OPC Import, SX-OPEN** and **EDE** must be reset in the **Cover** sheet.

If tables are missing, they might be hidden. Use the context menu below the table to unhide the tables.

3.2 Preparation - Checking global Excel settings



To avoid problems when using SX Configurator with different language settings always use a **dot** as decimal separator and a **comma** as thousand separator.

3.3 Cover sheet – Entering the project data

Here, you enter specific information about the project to describe it in full. The worksheet is described in detail in section 5.1.

Action	Step	Row	Procedure
Project definition	**1**	**PROJECT NAME**	Required → Define
	2	**REVISION**	Required → Define
	3	**Rows 4 to 7, 9 to 11**	Optional → Define
	4	**OPTIONS** section	Optional → Define

3.4 Importing data

Section 4 contains all the background information associated with the import of OPC data.

Action	Step	Menu/Button	Procedure
Select the relevant options	1	**Get OPC data**	Opens the dialog box for importing OPC data. Select the type and source of the OPC import, as described in section 4.
Select the type of data import	2	**Full/Based on device/Number of fields**	Enable the required type of import. Import the OPC server with a **Full** import
Select import location, online/offline	3a	**File**	Import from a file containing the OPC data in the appropriate format (offline).
	3b	**Local or Remote Computer**	If there is a connection to the server, import directly from the OPC server (online)
BACnet names	4	**Suggest BACnet Object Name and Keyname**	Specify whether the name of the BACnet object is to be derived automatically from the OPC name.
Start import	5	**OK**	Click to start import.

3.5 Checking the OPC server

Sheet	Step	Procedure
OPC server	1	Check the sheet that was automatically filled in during the import operation (see also section 5.5)

3.6 Entering the BACnet device

Sheet	Step	Procedure
BACnet device	1	Define the required BACnet devices (see section 5.6)

3.7 Editing the main configuration sheet

The worksheet and individual values are described in detail in section 5.2.

Following the import of the OPC data, the OPC section is already completed. To complete the SX configuration, the remaining data must now be added.
All the imported OPC items must now be assigned to the appropriate BACnet objects and properties in the BACnet section of the sheet.

For each mapping rule (= one row), other, optional functions such as alarm, trend or mapping functions can be assigned and enabled.

3.7.1 Completing the OPC section

Column	Step	Procedure
OPC Access Rights		User inputs in this field are ignored
OPC Data type		User inputs in this field are ignored
VT_ARRAY Index	1	If a value exists as a VT_Array, specify which index of the value array is to be mapped.
OPC Server Ref (required)	2	The reference to the relevant OPC server is automatically imported from the **OPC Server** sheet.
OPC Group Update Rate (optional)	3	Overwrite individual items as necessary. The default update rate is defined for each server in the **OPC Server** sheet. If more than one OPC group with the same update rate is required, enter update rates which differ by 1: e.g. 800, 801, 802 etc.
Item Set Ref (optional)	4	Enter the reference from the **Item Set** sheet
Device Set Ref (optional)	5	Enter the reference from the **Device Set** sheet

3.7.2 Completing the BACnet section

Column	Step	Procedure
BACnet Device Ref (required)	**1**	In the **BACnet Device** sheet, enter the reference to the BACnet device to which the BACnet object is be assigned
BACnet Object Type (required)	**2**	Select the object type for mapping values on the BACnet side (**Objects** sheet)
Object Instance (required)	**3**	Select an object instance.
Key name (required)	**4**	(= User Designation). Automatically entered in the data import process (normally identical to the BACnet object name)
BACnet Object Name (required)	**5**	(= Technical Designation). Automatically entered in the data import process if this option is enabled.
Description (optional)	**6**	Enter an object description, if required.
BACnet Property (required)	**7**	Enter a reference to the BACnet property to which the OPC value is to be mapped (**Properties** sheet). If blank, [PrVal] is entered.
Destination (optional)	**8**	Specify the destination of changes of value for all properties not related to [PrVal].
Force Update (optional)	**9**	Enter as applicable.
Relinquish Default (optional)	**10**	Enter a value for relinquish-default.

3.7.3 Completing BACnet information for analog objects

Column	Step	Procedure
BACnet unit (optional)	**1**	Enter a reference number or a shortcut from the **Unit** sheet.
Min Present Value (optional)	**2**	Enter the floating point number which is the minimal value which can be delivered from OPC item.
Max Present Value (optional)	**3**	Enter the floating point number which is the maximum value which can be delivered from OPC item.
COV increment (optional)	**4**	Enter a value for COV increment.
Deadband (optional)	**5**	Enter a value for deadband.
Resolution (optional)	**6**	Enter a value for resolution.
Update Interval (optional, analog-input only)	**7**	Enter a value for update interval.

3.7.4 Completing BACnet information for binary objects

Column	Step	Procedure
Polarity (optional)	1	Enter 0 for normal polarity and 1 for inverse polarity, binary-input and binary-output only.
Active Text (optional)	2	Enter the state text for present-value 1.
Inactive Text (optional)	3	Enter the state text for present-value 0.
Min Off Time (optional)	4	Enter the minimum off time.
Min On Time (optional)	5	Enter the minimum on time.

3.7.5 Completing BACnet information for multistate objects

Column	Step	Procedure
No of States (required)	1	Insert the number of states.
State Text Ref	2	Insert a reference to **State** sheet.

3.7.6 Completing the BACnet alarm info section

<table>
<tr><th>Column</th><th>Step</th><th colspan="2">Procedure</th></tr>
<tr><td rowspan="4">High Limit
(analog objects only)</td><td rowspan="4">1</td><td colspan="2">Enter as applicable:</td></tr>
<tr><td>Blank</td><td>= will be filled with max present value by SX Configurator</td></tr>
<tr><td>Value</td><td>= High alarm limit</td></tr>
<tr><td>x</td><td>= Alarm limit from OPC item</td></tr>
<tr><td rowspan="4">Low Limit
(analog objects only)</td><td rowspan="4">2</td><td colspan="2">Enter as applicable:</td></tr>
<tr><td>Blank</td><td>= will be filled with min present value by SX Configurator</td></tr>
<tr><td>Value</td><td>= Low alarm limit</td></tr>
<tr><td>x</td><td>= Alarm limit from OPC item</td></tr>
<tr><td rowspan="3">Alarm Value(s)
(binary objects)</td><td rowspan="3">3a</td><td colspan="2">Enter as applicable:</td></tr>
<tr><td>0</td><td>= Alarm for present-value 0</td></tr>
<tr><td>1</td><td>= Alarm for present-value 1</td></tr>
<tr><td rowspan="2">Alarm Value(s)
(multistate objects)</td><td rowspan="2">3b</td><td>Blank</td><td>= No alarm</td></tr>
<tr><td colspan="2">One or more values from 0...127, separated with commas (e.g. 2,5,11 etc.)</td></tr>
<tr><td rowspan="3">Fault Values
(multistate objects)</td><td rowspan="3">4</td><td colspan="2">Enter as applicable:</td></tr>
<tr><td>Blank</td><td>= No alarm</td></tr>
<tr><td colspan="2">One or more values from 0…127, separated with commas (e.g. 2,5,11 etc.)</td></tr>
<tr><td>Notification Class
(optional)</td><td>5</td><td colspan="2">Enter a class-number which is configured in NC sheet, if left blank notification-class 31 will be filled in by SX Configurator.</td></tr>
<tr><td>Message Text
(optional)</td><td>6</td><td colspan="2">Define text for each individual alarm → Strongly recommended!</td></tr>
</table>

Column	Step	Procedure
Notify Type (optional)	**7**	Enter as applicable:
		0 = Alarm
		1 = Event
Event Enable (optional)	**8**	Enter as applicable (addition of 3 event enable bits):
		1 = to-offnormal enabled
		2 = to-fault enabled
		4 = to-normal enabled
Time Delay (optional)	**9**	Enter delay time

Important For binary objects there may be one number only in column Alarm Value(s)

There is no check in SX Configurator for the same value used in Alarm Value(s) and Fault Value(s) of a multistate object.

3.7.7 Completing the BACnet trend info section

Column	Step	Procedure
Trend Log Object Instance (optional)	**1**	Enter the instance number of the trend log BACnet object

3.7.8 Completing the BACnet schedule info section

Column	Step	Procedure
Schedule Object Instance (optional)	**1**	Enter the instance number of the schedule BACnet object

3.7.9 Completing the alarm suppression section

Column	Step	Procedure
Propagation Delay (optional)	**1**	Delay time before propagating fault states to SX BACnet server
Subsystem Status (optional)	**2**	OPC Item describing communication status of appropriate subsystem
Subsystem Value (optional)	**3**	Item value of subsystem status which represents a valid communication state

3.7.10 Completing the function info section

Column	Step	Procedure	
Avoid Default Mapping (optional)	**1**	Disable default mapping of alarms	
Uses OPC Quality (optional)	**2**	Blank or <>1	= Use item value
		1	= Use item status
Function Name (optional)	**3**	Enter function name from **Functions** sheet	
Parameters (1…40) (optional)	**4**	Define other input parameters (according to function)	

3.8 Editing other sheets

Proceed in the order shown below:

Sheet	Step	Procedure
OPC server	**1**	Before editing the main **Configuration** sheet: Check the sheet that was automatically completed during the import process (see also section 5.5)
BACnet Device	**2**	Before editing the main **Configuration** sheet: Define the required BACnet devices (see also section 5.6)
Unit (Optional)	**3**	Define engineering units more simply or more briefly if necessary (see also section 5.7)
State (required, if multistate objects used)	**4**	Define state identifiers for multistate objects (see also section 5.8)
NC (Optional)	**5**	Define additional notification classes if necessary (see also section 5.2.6)
BSTR Hash (Optional)	**6**	Define hash values for strings (see also section 5.13)
Item Set & **DeviceSet** (Optional)	**7**	Identify possible item sets or device sets. If these sets can be identified, they must be defined once (see also sections 5.3 and 5.4).
Trend (Optional)	**8**	When compiling configuration a trendlog object will be created for every object in configuration sheet with a trendlog object instance. After this compilation the objects in **Trend** sheet can be modified, e.g. changing the default object names (see also section 5.2.7).
Schedule	**9**	When compiling configuration a schedule object will be created for every object in configuration sheet with a schedule object instance. After this compilation the objects in **Schedule** sheet can be modified, e.g. assigning calendar object to exception schedule (see also section 5.2.8).

3.9 Re-importing the OPC data

Re-import the OPC data online or offline, if sets are defined in the **Item Set** and **Device Set** sheets. Depending on the situation, select either **Based on Device** or **Number of fields** as type of import (see section 4.2).

3.10 Compiling the SX configuration

The SX configuration is completed after creation of the downloadable SX Open configuration files (CSV files) and the BACnet object descriptions by compiling the data.

Create SX-OPEN files

The SX configuration process is started by use of the illustrated button. The process is not complete until the compilation runs without errors. Any errors must be eliminated before repeating the compilation.

The process is recorded in a log file. The configuration is only complete after an error-free compilation.

3.11 Completing Trend configuration

Adjust properties in **Trend** sheet if necessary.

3.12 Completing Schedule configuration

Adjust properties in **Schedule** sheet if necessary.

4 Importing OPC data

The data objects available on the OPC server (primarily the server object and the various OPC items with name, data type, type of access etc.) are defined by importing the relevant object description list, or if there a connection to the server is established, by interrogating the OPC server.

Note

In normal circumstances, SX Open and the OPC server are installed on the same hardware platform. With large volumes of data, it is advisable for reasons of performance to install these two components on separate PCs. To overcome DCOM limitations an OPC tunneler is strongly recommended in the distributed case.

4.1 OPC data sources

OPC data can be imported in two ways:

- **Online:**
  SX Configurator retrieves the data directly from a connected OPC server and saves it in an SX file.
- **Offline:**
  In the worst case, the object definitions must be copied and pasted from an existing OPC item list, or even entered again manually.
  SX Configurator reads the data either from a fixed-format file or from an offline CSV file from the Desigo Insight OPC import tool (only format V1.3 is supported).

Import from OPC server

The server data is imported in two stages:

1. The data is retrieved from the server and stored in a file
2. The stored data is imported from the file into SX Configurator

The saved data is stored locally (as a comma-separated values list) in a file named **<OPCServerName>_<Date-Time>.sx** after which it can be re-imported at any time, with or without a connection to the OPC server.
If the configuration of SX Open is to be continued on another PC, it is advisable to copy not only the **SX Configurator.xls** file, but also the ***.sx** file and, in the case of the OPC server MK8000 (Cerberus), the configuration file ***.crb**.

SX Offline

The ***.sx** file contains a header:

`OPCServerName,ComputerName(Optional),`**`full`**

This is followed by rows containing:

```
"OPCItem",OPCItemVariantType,AccessRights,"OPCDescription","Unit"
```

`"OPCItem"`
`"OPCDescription"`
`"Unit"`
	Write between inverted commas

`OPCItemVariantType`	Must be a variant (VT_BOOL, VT_I2, VT_R4, …);
AccessRights	R, RW or W

4.2 OPC data import types (Import Type)

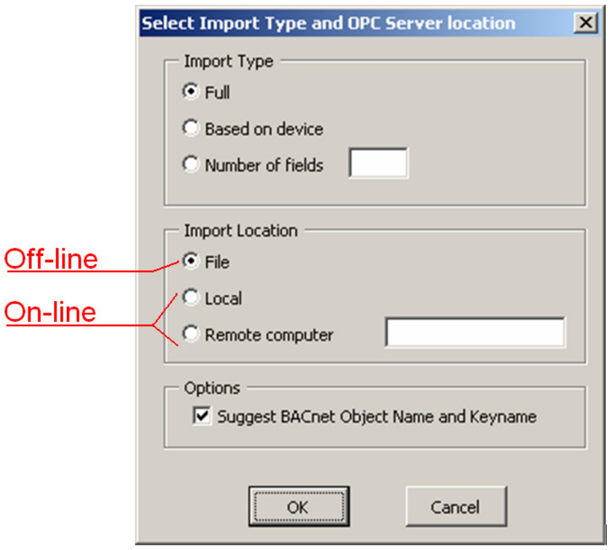


Figure 4-1

Note

The **Full** import type must always be used at least once if no data has yet been imported. After this, an **OPC server reference** is generated, and the item sets or device sets can be created:

- Analyze the data and decide whether **Number of fields** or **Based on device** are applicable.
- If so, copy the relevant OPC item data from the **Configuration** sheet to the **Item Set** sheet, modify the data and add the necessary information including BACnet data.
- Re-import the data from the file generated during the first import, using the appropriate import type.

If more than one OPC item is mapped to a BACnet object, these items must be directly one underneath the other, i.e. the rows must be sorted by BACnet object.

4.2.1 Full

All OPC items are displayed in full in the **Configuration** sheet, and the data can be edited as required.

[PrVal]

With a **Full** import, the default BACnet property is [PrVal] for all defined points.

STOP **Note**

It is advisable to set **Warning Level 2**, as otherwise, all the data will be checked.

Examples for a log report:

<table>
<tr>
<td>
SX-OPEN Configurator V2.00.025 - 10.11.09

Project Name: DemoConfiguration
SX-OPEN Compilation started on 16-Nov-2009, 07:21:12
SX-OPEN files generated in C:\Documents and Settings\All Users\Application Data\Siemens\SX-Open\Config
EDE files generated in C:\Documents and Settings\All Users\Application Data\Siemens\SX-Open\Config
Warning Level is 1
Using default OPC QUALITY mapping

Configuration compilation failed; output files invalid...

SX-OPEN Compilation finished on 16-Nov-2009, 07:21:12

Generated error/warning messages:
================================
Warning Row 6: Invalid OPC item name as Subsystem Status ignored:
Warning Row 7: Invalid OPC item name as Subsystem Status ignored:
ERROR Row 8: BACnet object (SXOPEN'Frame'Status) already exists for BACnet device instance 255
ERROR Row 9: BACnet object (SXOPEN'Frame'Status) already exists for BACnet device instance 255
Warning Row 10: Invalid OPC item name as Subsystem Status ignored:
Warning Row 11: Invalid OPC item name as Subsystem Status ignored:
Warning Row 12: Invalid OPC item name as Subsystem Status ignored:
Warning Row 13: Invalid OPC item name as Subsystem Status ignored:
Project Name: DemoConfiguration
SX-OPEN Compilation started on 16-Nov-2009, 07:21:12
SX-OPEN files generated in C:\Documents and Settings\All Users\Application Data\Siemens\SX-Open\Config
EDE files generated in C:\Documents and Settings\All Users\Application Data\Siemens\SX-Open\Config
Warning Level is 1
Using default OPC QUALITY mapping

Configuration compilation failed; output files invalid...

SX-OPEN Compilation finished on 16-Nov-2009, 07:21:12

Generated error/warning messages:
================================
Warning Row 6: Invalid OPC item name as Subsystem Status ignored:
Warning Row 7: Invalid OPC item name as Subsystem Status ignored:
ERROR Row 8: BACnet object (SXOPEN'Frame'Status) already exists for BACnet device instance 255
ERROR Row 9: BACnet object (SXOPEN'Frame'Status) already exists for BACnet device instance 255
Warning Row 10: Invalid OPC item name as Subsystem Status ignored:
Warning Row 11: Invalid OPC item name as Subsystem Status ignored:
Warning Row 12: Invalid OPC item name as Subsystem Status ignored:
</td>
<td>
SX-OPEN Configurator V2.00.025 - 10.11.09

Project Name: DemoConfiguration
SX-OPEN Compilation started on 16-Nov-2009, 07:18:40
SX-OPEN files generated in C:\Documents and Settings\All Users\Application Data\Siemens\SX-Open\Config
EDE files generated in C:\Documents and Settings\All Users\Application Data\Siemens\SX-Open\Config
Warning Level is 1
Using default OPC QUALITY mapping

Configuration compilation successful... Warnings were generated; please, see below.

27 BACnet objects generated
1 BACnet device(s) generated
Generated file C:\Documents and Settings\All Users\Application Data\Siemens\SX-Open\Config\SXBacNet.csv
Generated file C:\Documents and Settings\All Users\Application Data\Siemens\SX-Open\Config\SXAlarmText.csv
Generated file C:\Documents and Settings\All Users\Application Data\Siemens\SX-Open\Config\SXBacNetStateText.csv

0 mapping entries generated
Generated file C:\Documents and Settings\All Users\Application Data\Siemens\SX-Open\Config\SXMapping.csv. Warnings were generated; please, see below.

0 OPC items generated
Generated file C:\Documents and Settings\All Users\Application Data\Siemens\SX-Open\Config\SXOPCClient.csv

0 Trend object(s) generated

27 EDE BACnet objects generated
9 EDE states generated
0 EDE units generated
Generated file C:\Documents and Settings\All Users\Application Data\Siemens\SX-Open\Config\SxEde.csv
Generated file C:\Documents and Settings\All Users\Application Data\Siemens\SX-Open\Config\SxEdeStateTexts.csv
Generated file C:\Documents and Settings\All Users\Application Data\Siemens\SX-Open\Config\SxEdeUnitTexts.csv

0 BACnet client mapping entries generated

Supervisor information: Generated file C:\Documents and Settings\All Users\Application Data\Siemens\SX-Open\Config\SXSupervisor.ini

DIEPT not executed...

SX-OPEN Compilation finished on 16-Nov-2009, 07:18:44

Generated error/warning messages:
================================
Warning Row 6: Invalid OPC item name as Subsystem Status ignored:
Warning Row 7: Invalid OPC item name as Subsystem Status ignored:
Warning Row 8: Invalid OPC item name as Subsystem Status ignored:
Warning Row 9: Invalid OPC item name as Subsystem Status ignored:
Warning Row 10: Invalid OPC item name as Subsystem Status ignored:
Warning Row 11: Invalid OPC item name as Subsystem Status ignored:
Warning Row 12: Invalid OPC item name as Subsystem Status ignored:
Warning Row 13: Invalid OPC item name as Subsystem Status ignored:
Warning The mapping-file is empty, probably you have only configured the supervisor-server: BACnet Device 255
</td>
</tr>
</table>

Export canceled

Export succeeded

If all the data is valid, the files are created in the folders specified in the **Cover** sheet (**SX-OPEN** and **EDE**).

4.2.2 Based on device

This import type can be used when there are several devices of the same type to import, e.g. room controller devices in a 400-room hotel. Assuming that 400 rooms generate the same type of information, a device template, the "Device Set", can be created. In this case, only the OPC item fields to the left (normally the Device ID) are displayed in the **Configuration** sheet. All other fields are in the **Device Set** sheet.

4.2.3 Number of fields

In some situations, OPC items follow a regular structure and can be mapped on the basis of the last field to specific BACnet properties in the same object.

```
RSAcc test.QC0101-NRUD (1).VC01-FC20.RSUAO1.Val
RSAcc test.QC0101-NRUD (1).VC01-FC20.RSUAO1.Man
RSAcc test.QC0101-NRUD (1).VC02-FC22.RSUAO2.Val
RSAcc test.QC0101-NRUD (1).VC02-FC22.RSUAO2.Man
RSAcc test.QC0101-NRUD (1).VC03-FC18.RSUAO3.Val
RSAcc test.QC0101-NRUD (1).VC03-FC18.RSUAO3.Man
```

In this case, only the OPC item fields to the left are displayed in the **Configuration** sheet. All other fields are in the **Item Set** sheet.
Each point has a value and a status. In this case, to simplify the data import, the entries in the **Item Set** sheet can be defined as follows:
For the OPC data import, select **Number of Fields** as the type of import, and **4** as the number of fields to be retained:

```
RSAcc test.QC0101-NRUD (1).VC03-FC18.RSUAO3.Val
----1----- -----2--------- -----3--- ---4–
```

When the data is re-imported, the field **Item Set Ref** and **Object Type** are filled in automatically, based on the entries in the **Item Set** sheet. The OPC item does not show the fields to the right (red in the illustration below).

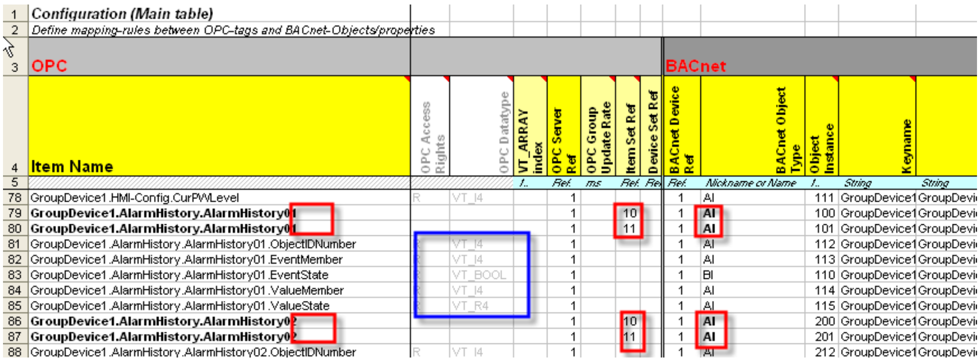


Figure 4-2

For items that are not found in the **Item Set** sheet (blue in the illustration), the **Item Set Ref** column is blank, and the associated **OPC Access Rights** and **OPC Datatype** are supplied as user information.

The **OPC Group Update Rate** field defines the OPC groups and the associated update rate.

Device Set Ref is not used for this type of import.

The **Item Set** sheet has a special use in cases where BACnet properties other than [PrVal] are to be mapped to or from an OPC item.

The idea underlying these import types is to offer a certain degree of automation for repetitive tasks, and to reduce the information in the **Configuration** sheet.

The only type of import that is mandatory is **Full.**

For an online import, the OPC server can be located on a different computer. In this case, the name of the remote computer should be entered.

4.3 Import Location

File

Clicking the **File** button displays a dialog box from which you can select and open the file.

Local or remote computer

If you select an online import via **Local** or **Remote computer**, the system searches for the local or remote computer and displays the available servers from which you can select the OPC server.

4.4 Options

If the check box **Suggest BACnet Object Name and Keyname** is selected, the name and key name of the BACnet object are created automatically from the OPC item string. for repetitive tasks.

Selecting BACnet objects based on OPC item type

SX Configurator uses builtin rules to assign BACnet object type based on OPC data type and OPC access rights. These rules can be adapted by changing hidden sheet **OPCImportTypeMapping**.

4.5 Change management

To simplify mapping-changes whenever OPC data is imported, whether via a file or online, the OPC items change color as follows:

OPC item color	Status of row	Comment
Black	OK	
Green	New OPC item	
Red	OPC item deleted	Not exported to SX files, even if additional BACnet information is available (warning is always generated)
Blue	BACnet data modified	Shows that information about the **Number of fields** import type or the **OPC import tool file source** has changed on the BACnet side.

Example

	A	B	C	D	E	F	G	H	I	J	K	L
1	*Configuration (Main table)*											
2	*Define mapping-rules between OPC-tags and BACnet-Objects/properties*											
3	OPC								BACnet			
4	**Item Name**	OPC Access Rights	OPC Datatype	VT_ARRAY index	OPC Server Ref	OPC Group Update Rate	Item Set Ref	Device Set Ref	BACnet Device Ref	BACnet Object Type	Object Instance	
5				*I..*	*Ref.*	*ms*	*Ref.*	*Re*	*Ref.*	*Nickname or Name*	*I..*	*String*
6	SERVER_STATE.State	R	VT_BOOL		1				1	BI	10	sbtrccv20
7	GroupDevice1.LocalMSGCL	R	VT_I4		1				1	AI	1000	GroupDevi
8	GroupDevice1.LocalFault	R	VT_BOOL		1				1	BI	20	GroupDevi
9	GroupDevice1.LocalGroup	R	VT_I4		1				1	AI	1001	GroupDevi
10	GroupDevice1.LocalNewMSGCL	R	VT_I4		1				1	AI	1002	GroupDevi
11	GroupDevice1.HierarchyMSGCL	R	VT_I4		1				1	AI	1004	GroupDevi
12	GroupDevice1.HierarchyFault	R	VT_BOOL		1				1	BI	40	GroupDevi
13	GroupDevice1.HierarchyGroup	R	VT_I4		1				1	AI	1005	GroupDevi
14	**GroupDevice1.AcknowledgeMSGCL**	RW	VT_I4		1				1	AI	1008	GroupDevi
15	GroupDevice1.AcknowledgeGroup	RW	VT_I4		1				1	AI	1009	GroupDevi
16	GroupDevice1.AcknowledgeAll	RW	VT_BOOL		1				1	MSI	100	GroupDevi
17	GroupDevice1.RCC-Config.TypeIP	RW	VT_I4		1				1	AO	100	GroupDevi
18	**GroupDevice1.RCC-Config.TypeIP200**								1	AO	101	GroupDevi
19	GroupDevice1.RCC-Config.WriteSetting	RW	VT_BOOL		1				1	BO	200	GroupDevi
20	GroupDevice1.RCC-Config.GivenIP	RW	VT_I4		1				1	AO	900	GroupDevi
21	**GroupDevice1.LocalNewFault**	R	VT_BOOL		1							GroupDevi
22	**GroupDevice1.LocalNewGroup**	R	VT_I4		1							GroupDevi
23	**GroupDevice1.HierarchyNewMSGCL**	R	VT_I4		1							GroupDevi
24	**GroupDevice1.HierarchyNewFault**	R	VT_BOOL		1							GroupDevi
25	**GroupDevice1.HierarchyNewGroup**	R	VT_I4		1							GroupDevi

Figure 4-3

Note

When **Create SX-OPEN files** is clicked, the colors **green** and **blue** revert to **black**.

The **red** items always generate the following warning:

```
**Warning** Row XX: Item does not exist anymore; object not exported...
```

until the rows concerned have been checked and removed.

New or modified items contain a "1" in the **ForceUpdate** column, to force SX Open to reset to the changed/new value.

4.6 Limitations

Currently, SX Open does **not** support the following data types:

- VT_SAFEARRAY
- VT_DISPATCH
- VT_ERROR
- VT_VARIANT
- VT_UNKNOWN
- VT_DECIMAL
- VT_VOID
- VT_HRESULT
- VT_PTR, VT_CY, VT_DATE
- Array of VT_BSTR

Important

To **improve performance** when large volumes of data are used on the same OPC server, separate OPC groups should be created, each with approx. 150...300 OPC items depending on the computer speed and memory.

It is possible to use **OPC items from more than one OPC server** if the selected import type is a **Full** import.

5 Worksheets in SX Configurator

SX Configurator is based on worksheets that contain data for the OPC/BACnet conversion.

The data is arranged in separate sheets to provide maximum support when configuring mass data. The basic principle is that specific items should be defined in detail once only in a sheet, with only a reference required in the main **Configuration** worksheet. Configuration sheet is furthermore designed in a way which allows an easy update of values from an existing system before compiling a new configuration. This enables frequently recurring routines to be dealt with more efficiently. Apart from the **Cover** sheet, the configuration data is entered row by row.

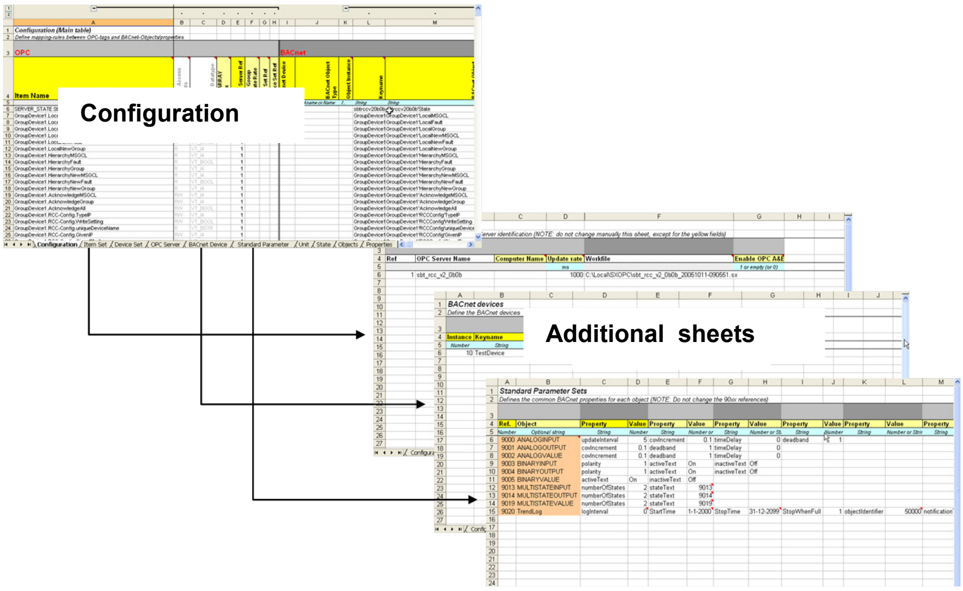


Figure 5-1

Worksheet overview		**Contents**	**Section**
Title sheet	**Cover**	Project description	**5.1**
Main sheet for configuration of OPC, BACnet, alarm and trend functions	**Configuration**	Main sheet with the OPC ↔ BACnet mapping rules	**5.2**
Sheets with data to which reference is made in the main sheet	**Item Set**	Summary of recurrent mapping groups, including BACnet devices, OPC servers, Standard Parameters, engineering units, status text etc.	**5.3**
	Device Set	Summary of recurrent device mappings	**5.4**
	OPC Server	Name and location of OPC servers	**5.5**
	BACnet Device	Definition of BACnet devices	**5.6**
	Unit	Admissible engineering units	**5.7**
	State	Admissible states for multistate objects	**5.8**
	Objects	Admissible BACnet object types	**5.9**
	Properties	Admissible BACnet properties	**5.10**
	Functions	Function block definitions	**5.11**
	BACnet-Client	Mapping rules when using the BACnet client	**5.12**

Worksheet overview		Contents	Section
	Nofification-Class	Definition of notification classes used in all configured BACnet devices.	**5.2.6**
	Hash-Values	Definition of hash-values for items of VT_BSTR type	**5.13**
	Trend	Definition of used trendlog objects	**5.2.7**
	Schedule	Definition of used schedule objects	**5.2.8**

Color code

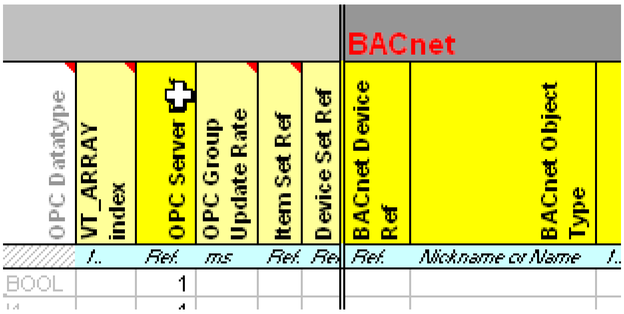


Figure 5-2

The column names show the type of information required in each field.

Object Instance	Yellow – normal columns with mandatory information
Force Update	Light yellow – optional columns, to be completed as necessary
OPC Access Rights	"Grayed out" print on a white background – generally only for extra information to help with configuration
String	Light turquoise – indicates the engineering unit in which the information is to be supplied
Ref.	(Light turquoise): References to sheets. These must be entered in a sheet in the form of an integer index to the relevant data record
Pres Value Range	Light yellow, dotted – Analog values only
Fault Values	Light yellow, hatched – Warning: Digital values only
1 ackRequired 0 ackedTransitions	Tan – Default values which must not be modified.

Further information

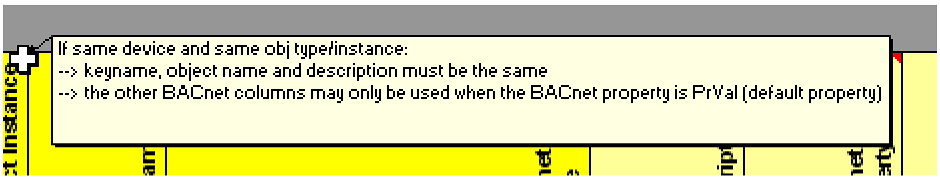


Figure 5-3

A red mark in the top right corner of a cell indicates that additional information is accessible via this cell. For access to this information, position the cursor directly over the red area, without clicking the mouse.

5.1 Cover sheet

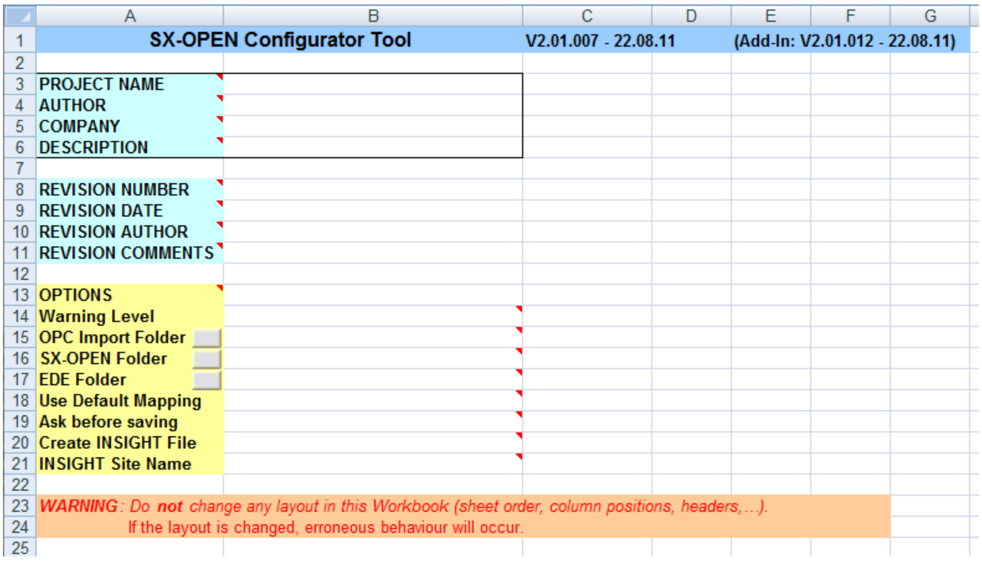


Figure 5-4

The **Cover** sheet contains basic project information, such as name, description and person(s) responsible, plus a section for general and optional settings and file storage information..

5.1.1 Administrative section

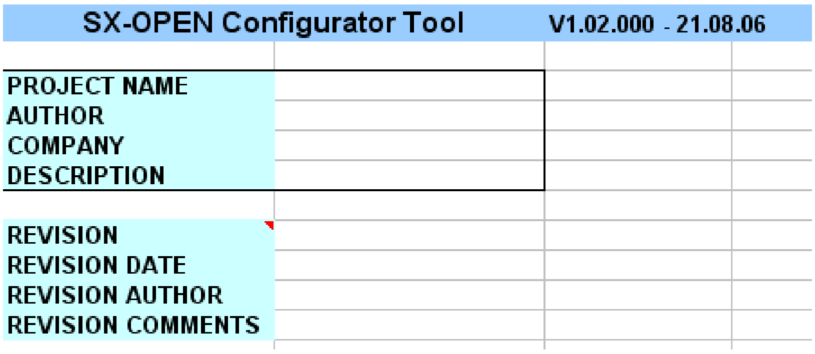


Figure 5-5

Fill in the administrative details (the light turquoise cells) to suit your requirements.

Note

Apart from the **PROJECT NAME and AUTHOR**, all entries are optional. The information is used for documentation and management of the configuration data and does not affect how SX Configurator behaves. Furthermore this information is needed if the integrated DIEPT is used.

PROJECT NAME, AUTHOR

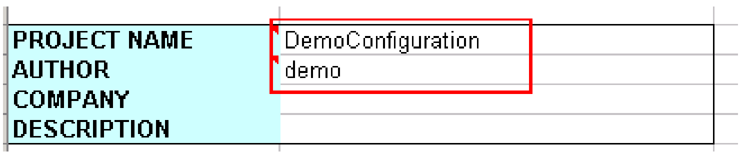


Figure 5-6

When the file is saved for the first time, the project name is automatically appended to the file name, making the file name unique.

Example

1. Complete the **PROJECT NAME** field, e.g. by entering the name "My Project".
2. When the file is saved, its name is automatically extended to:
   SX Configurator- My Project.xls

From now on, only the project-specific file is used. The original file remains unchanged.

If you try to save the file with an empty **PROJECT NAME** field, a warning is displayed.

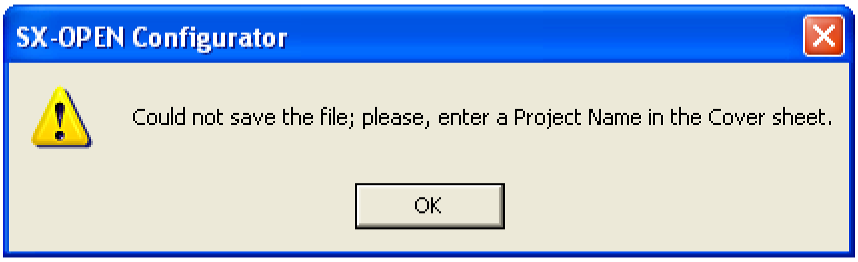


Figure 5-7

REVISION

Revision information must be entered in the columns to the right of the **REVISION** cells, e.g. if the original version information appears in rows 8 to 11 of column B, then information about the next revision is shown in rows 8 to 11 of column C, etc. The revision will be used to generate the BACnet property **database-revision** and must follow a format of type **<number>.<number>**. While generating database-revision, only the first three numbers after the decimal point will be used.

Example

1. Complete the **REVISION** field, e.g. by entering the value "1.001".
2. When configuration is compiled, database-revision will automatically set to: **1001**.

5.1.2 OPTIONS section

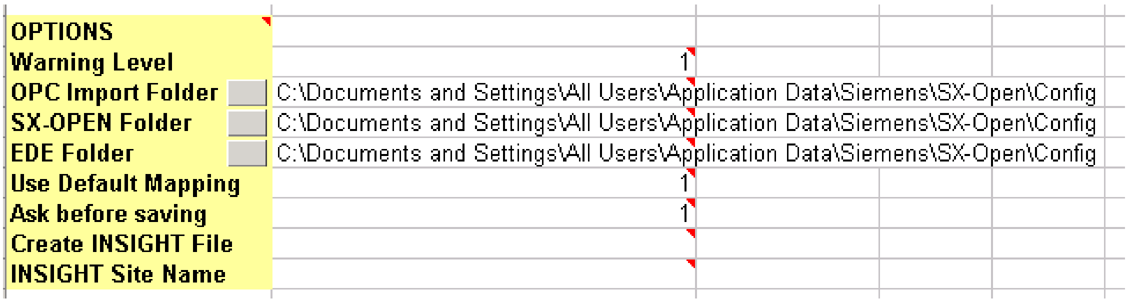
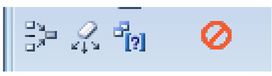


Figure 5-8

This section (light yellow fields) allows you to modify the general behavior of SX Configurator and define the paths to any additionally required files, or to the file location of the SX configuration. The default is the path via which **SX Configurator.xls** was opened.

Important

Once the file is saved by its project-specific name, the **OPC Import Folder, SX-OPEN Folder** and **EDE Folder** paths are no longer updated automatically.

Row	Function	Default
Warning Level	When an SX configuration is compiled, this opens a log file in which the process is documented. This is where information, warning and error messages are shown. **Warning Level** can be used to control the number of warning messages. The options are as follows: 0 = Warnings suppressed For situations which are not errors, but about which the user needs to be informed so that corrective action can be taken. 1 = Important warnings displayed 2 = All warnings displayed	1
OPC Import Folder	Inserted during the data import process. Path to access the file for the OPC items list for offline import into SX Configurator.	
SX-OPEN Folder	Inserted during the data import process. Path to the storage location for the various SX Open configuration files.	
EDE-Folder	Inserted during the data import process. Path to the storage location for the BACnet object descriptions in EDE and XML format.	
Use Default Mapping	For further information refer to section 5.1.3. 0 = No automatic mapping of the OPC item status to alarm properties of the BAC net object. 1 = OPC item status mapped to alarm properties of the BAC net object.	1
Ask before saving 	For safety, the Excel file is saved whenever the **Create SX-OPEN Files** button is clicked: 0 = Immediately 1 = Only after confirmation by the user. This allows the user to decide whether or not to save the data.	1

Row	Function	Default
Create INSIGHT File	For further information refer to [1] Blank = BACnet object description will not be created automatically as an XML file Not blank, e.g. 1 = XML object description created and saved in the location specified in the **EDE Folder**.	Blank
INSIGHT Site Name	For further information refer to [2] If the **Create INSIGHT File** field is enabled, the Desigo Insight site name must be entered here; the SX Open objects are arranged under this name.	Blank

[1] Create INSIGHT File

To allow the user of a BACnet client to operate with the BACnet objects generated by SX Open, SX Configurator always creates descriptions of the BACnet objects in EDE (Electronic Data Exchange) format, a format defined by B.I.G. EU, the BACnet Interest Group, Europe, for the electronic exchange of BACnet configuration data.

The **Create INSIGHT File** field is used to configure whether or not the BACnet object description should also be generated as an XML file. This XML description can be imported directly into Desigo Insight with the **DbImport** tool.

[2] INSIGHT Site Name

It is also critically important that the Desigo Insight site name reference is identical to the site name under which the SX BACnet objects are listed. This site must be set up on Desigo Insight before the import.

Important

SX BACnet objects must always be accommodated on separate sites.

It is not permissible to mix SX and PX objects on the same site.

Important

The xml file created by SX Configurator can only be used with Desigo Version 4.1 and higher For Desigo Version 3.0 must be used based on the created EDE files.

5.1.3 Default mapping

An OPC item consists of:

- Name
- Value
- Status
- Time stamp

Activating **Default Mapping** means that not only is the item value mapped to [PrVal] of the BACnet object, but also that the OPC status is mapped, more or less invisibly, to the relevant alarm properties of the same object. Depending on the OPC status, the alarm state on the BACnet side is then NORMAL or an OFFNORMAL or FAULT alarm.

Note

Even if **Default Mapping** is active, it can still be enabled or disabled individually for each value to be mapped, via the **Avoid Default Mapping** column of the **Configuration** sheet.

SX Open can generate default mappings based on the OPC Quality. This generates the following mapping information for each BACnet object defined as [PrVal]. The object information relates to **Reliability, Out_of_Service** and **Overridden in Status_Flags:**

OPC		**BACnet**		
Quality	**Substatus**	**Reliability**	**Out_Of_ Service**	**Overridden in Status_Flags**
0x00 Bad	0 Non-specific	7 Unreliable-other	False	False
0x04 Bad	1 Configuration error	2 Over range	False	False
0x08 Bad	2 Not connected	5 Shorted-loop	False	False
0x0c Bad	3 Device Failure	4 Open-loop	False	False
0x10 Bad	4 Sensor Failure	1 No-sensor	False	False
0x14 Bad	5 Last Known Value	5 Shorted-loop	False	False
0x18 Bad	6 Comm Failure	4 Open-loop	False	False
0x1C Bad	7 Out of Service	0 No-fault-detected	TRUE	False
0x40 Uncertain	0 Non-specific	7 Unreliable-other	False	False
0x44 Uncertain	1 Last usable value	6 No-output	False	False
0x50 Uncertain	4 Sensor not accurate	1 No-sensor	False	False
0x54 Uncertain	5 Eng. Units exceed.	8 Process-error	False	False
0x58 Uncertain	6 Sub-normal	7 Unreliable-other	False	False
0xC0 Good	0 Non-specific	0 No-fault-detected	False	False
0xD8 Good	6 Local	0 No-fault-detected	False	TRUE

The default mapping is generated only if **Use Default Mapping** in the **Cover** sheet is not blank, and only for objects for which the **Avoid Default Mapping** cell is left blank.

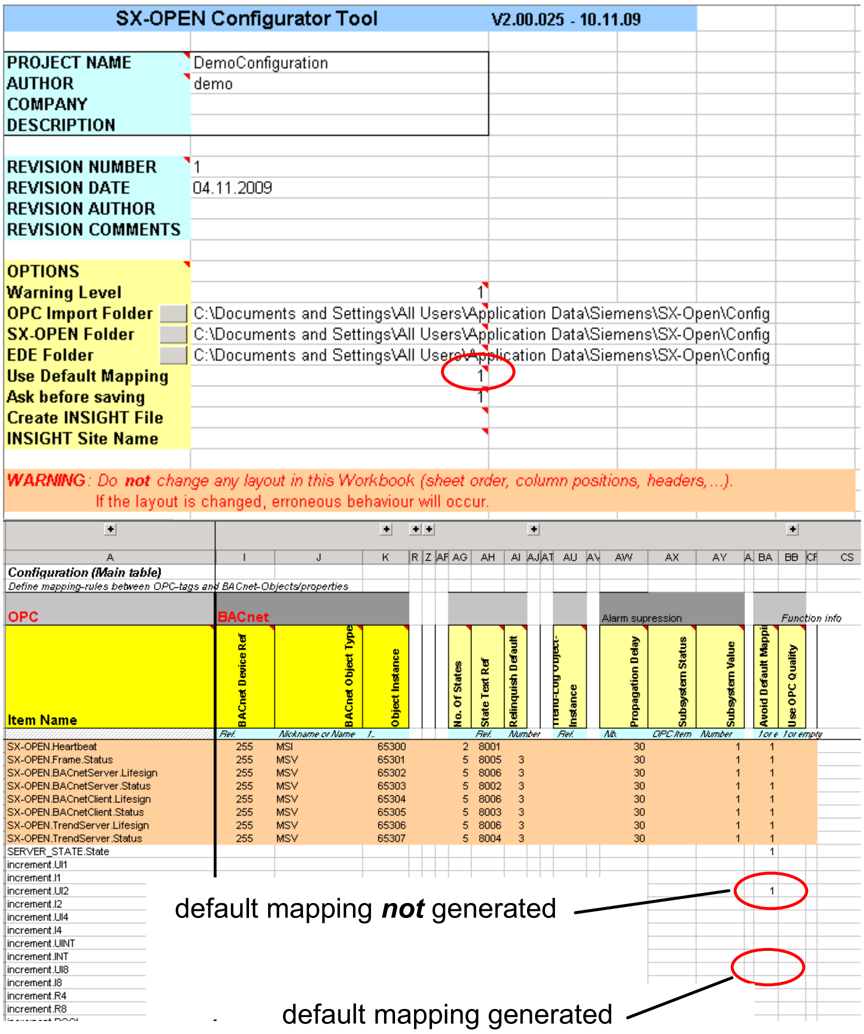


Figure 5-9

This default mapping generates **error events** based on the sheet above, using the configured notification class.

5.2 Main Configuration sheet

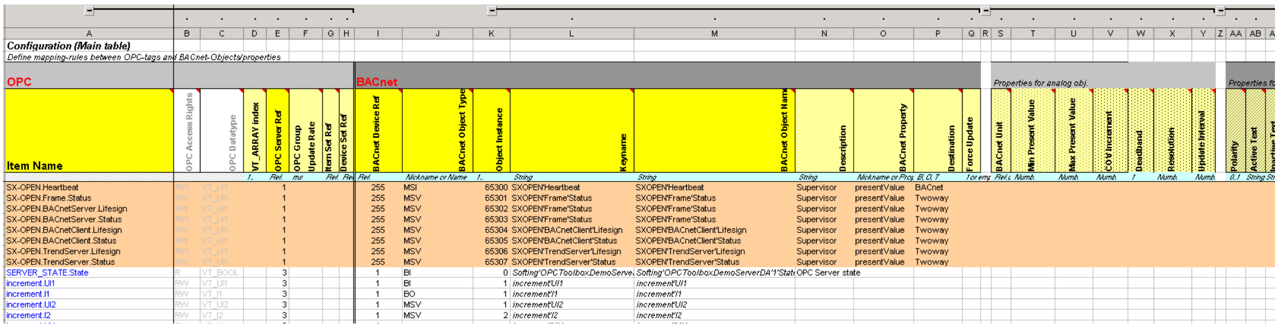


Figure 5-10

The **Configuration** sheet contains the actual SX Open configuration with the rules for mapping OPC data to BACnet objects and properties and the use of optional system functions such as alarm and trend. This is also where the use of specific function blocks for mapping signals and functions should be configured (for further information on this subject, refer to the Engineering Guideline.

The obvious approach is a complete OPC data import: all OPC item strings are imported from the selected source into the **Configuration** sheet (first column). There are no further actions apart from the automatic insertion of the OPC server information in the **OPC Server** sheet, and the **OPC Server Ref** in the **Configuration** sheet.

Mapping rules

SX Open is basically configured by assigning an OPC item (variable name, value, status and time stamp) to one or more properties of a BACnet object (= mapping rule). Each row in the **Configuration** sheet contains a mapping rule of this type.

In addition to this basic configuration, further optional functions can be enabled and configured in other columns.

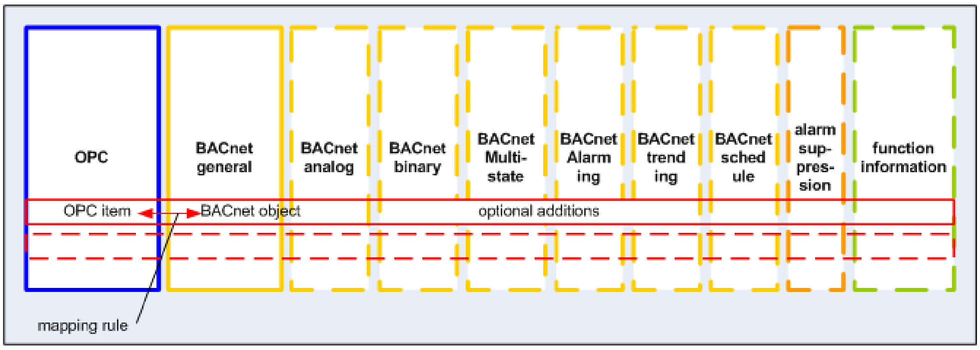


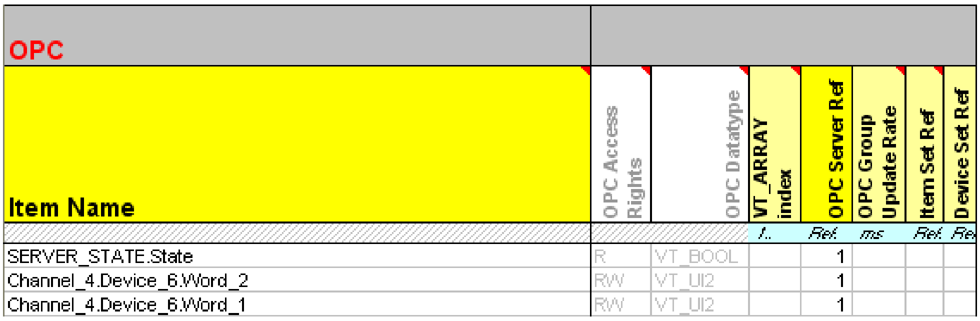
Figure 5-11

Structure

The sheet is divided into the following sections:

- OPC
- BACnet
    - general settings
    - analog objects settings
    - binary objects settings
    - multistate objects settings
    - alarm information
    - trend information
    - schedule information
- Alarm suppression
- Function information

5.2.1 OPC area



OPC								
Item Name	OPC Access Rights	OPC Datatype	VT_ARRAY index	OPC Server Ref	OPC Group Update Rate	Item Set Ref	Device Set Ref
			I..	Ref.	ms	Ref.	Re
SERVER_STATE.State	R	VT_BOOL		1			
Channel_4.Device_6.Word_2	RW	VT_UI2		1			
Channel_4.Device_6.Word_1	RW	VT_UI2		1			

Figure 5-12

Column	Description
Item Name	Name of the OPC item. The item contains a value, a status and a time stamp for the last change of value.
OPC Access Rights	For information only (user inputs in this field are ignored). The input serves an aid to the definition of other fields (refer also to the Engineering Guideline [1]).
OPC Data Type	As above
VT_ARRAY Index	SX Open can only map one value per row. Define here which index of a value array is to be mapped.
OPC Server Ref	To facilitate the import of data from the OPC server with its own unit strings, this column must contain the string for mapping to the BACnet units (value from the **OPC Server** sheet and **Ref** row, for the required OPC server).
OPC Group Update Rate	Update rate in milliseconds for all OPC items of a given OPC server. In SX Open, the update rate defines the OPC group for the OPC server concerned. If this column is not blank in the **Configuration** sheet, the value (update rate in ms) is used, and a new OPC group is created. Otherwise the default value from the **OPC Server** sheet is used, and just one OPC group is created. The update rate can be overwritten for each item in the **Configuration** sheet. All items with the same update rate are always grouped into one item group and retrieved simultaneously. The overall performance of SX Open can be drastically impaired if too many items in a group have to be retrieved at the same time
STOP	If you need to define more than one OPC group with the same update rate, enter update rates which differ by 1: e.g. 800, 801, 802 etc.

Column	Description
Item Set Ref	Reference to an **Item Set** from the sheet of the same name. Can be used for this type of import.
Device Set Ref	Reference to a **Device Set** from the sheet of the same name. Not used for this type of import.

STOP **Note**

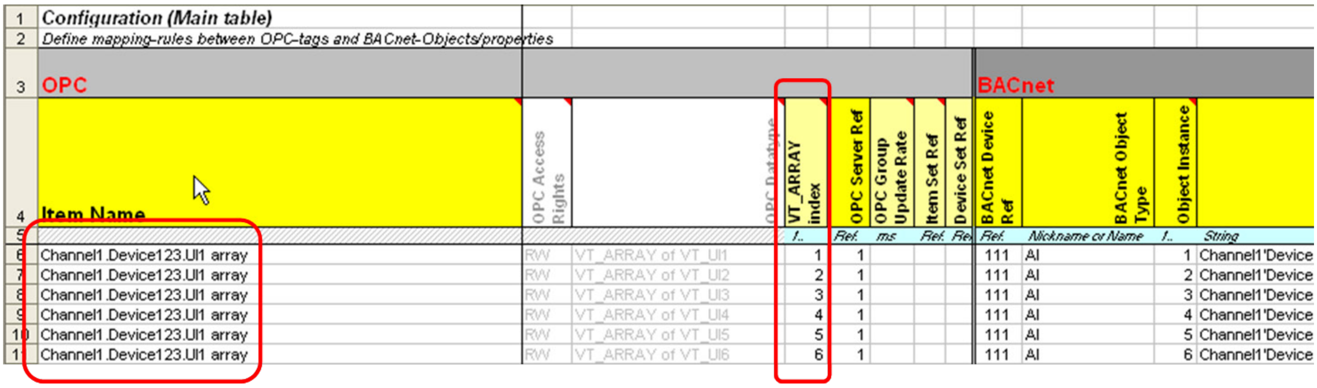
Depending on the number of items, the number of changes-of-value per unit of time and the use of the values on the BACnet side, it is therefore advisable to select update rates that are as long as possible, to avoid unnecessarily impairing the performance of SX Open. If several groups are to be formed with the same update rate, the update rates entered must vary by milliseconds, e.g. 1000ms, 1001ms, 1002ms, etc.

VT_ARRAYs

For OPC items which are VT_ARRAYS, each element of an array (matrix) uses one row in the **Configuration** sheet.

SX Open supports only a fixed number of elements per VT_ARRAY.

For data imported from the OPC source, only the first row of the VT_ARRAY is displayed. If other elements of the array are to be mapped, copy the row and modify the array index and BACnet object name and keyname accordingly.



1	*Configuration (Main table)*											
2	*Define mapping-rules between OPC-tags and BACnet-Objects/properties*											
3	OPC								BACnet			
4	Item Name	OPC Access Rights	OPC Datatype	VT_ARRAY index	OPC Server Ref	OPC Group Update Rate	Item Set Ref	Device Set Ref	BACnet Device Ref	BACnet Object Type	Object Instance	
5				*I..*	*Ref.*	*ms*	*Ref.*	*Re*	*Ref.*	*Nickname or Name*	*I..*	*String*
6	Channel1.Device123.UI1 array	RW	VT_ARRAY of VT_UI1	1	1				111	AI	1	Channel1'Device
7	Channel1.Device123.UI1 array	RW	VT_ARRAY of VT_UI2	2	1				111	AI	2	Channel1'Device
8	Channel1.Device123.UI1 array	RW	VT_ARRAY of VT_UI3	3	1				111	AI	3	Channel1'Device
9	Channel1.Device123.UI1 array	RW	VT_ARRAY of VT_UI4	4	1				111	AI	4	Channel1'Device
10	Channel1.Device123.UI1 array	RW	VT_ARRAY of VT_UI5	5	1				111	AI	5	Channel1'Device
11	Channel1.Device123.UI1 array	RW	VT_ARRAY of VT_UI6	6	1				111	AI	6	Channel1'Device

Figure 5-13

5.2.2 BACnet section

BACnet								
BACnet Device Ref	BACnet Object Type	Object Instance	Keyname	BACnet Object Nam	Description	BACnet Property	Destination	Force Update
Ref.	Nickname or Name	I..	String	String	String	Nickname or Proj	B, O, T	I or emj
255	MSI	65300	SXOPEN'Heartbeat	SXOPEN'Heartbeat	Supervisor	presentValue	BACnet	
255	MSV	65301	SXOPEN'Frame'Status	SXOPEN'Frame'Status	Supervisor	presentValue	Twoway	
255	MSV	65302	SXOPEN'Frame'Status	SXOPEN'Frame'Status	Supervisor	presentValue	Twoway	
255	MSV	65303	SXOPEN'Frame'Status	SXOPEN'Frame'Status	Supervisor	presentValue	Twoway	
255	MSV	65304	SXOPEN'BACnetClient'Lifesign	SXOPEN'BACnetClient'Lifesign	Supervisor	presentValue	Twoway	
255	MSV	65305	SXOPEN'BACnetClient'Status	SXOPEN'BACnetClient'Status	Supervisor	presentValue	Twoway	
255	MSV	65306	SXOPEN'TrendServer'Lifesign	SXOPEN'TrendServer'Lifesign	Supervisor	presentValue	Twoway	
255	MSV	65307	SXOPEN'TrendServer'Status	SXOPEN'TrendServer'Status	Supervisor	presentValue	Twoway	
1	BI	0	*Softing'OPCToolboxDemoServe*	*Softing'OPCToolboxDemoServerDA'1'Stat*	OPC Server state			
1	BI	1	*increment'UI1*	*increment'UI1*				
1	BO	1	*increment'I1*	*increment'I1*				
1	MSV	1	*increment'UI2*	*increment'UI2*				
1	MSV	2	*increment'I2*	*increment'I2*				

Figure 5-14

The BACnet information has to be entered manually to define each mapped OPC item.

In the simplest case, some (or all) OPC items are mapped to [PrVal] of the BACnet objects.

This is the case if the following conditions apply to each OPC-item row in the **Configuration** sheet:

- **Item Set Ref** is blank
- **BACnet Device Ref**erence exists
- **Object Type** is defined
- **Object Instance** has been entered
- **Keyname** exists
- **Object Name** exists

Column	Description
BACnet Device Ref	Reference to a specific BACnet device in the **BACnet Device** sheet to which the BACnet object is be assigned There are as many BACnet devices as required. For further information, see [1)]
BACnet Object Type	Type of BACnet object. See the **Objects** sheet, section 5.9. Select an appropriate object for value mapping on the BACnet side in accordance with the OPC-side data type and access rights. Normally, integers and real values are mapped to analog, binary and multistate objects, and read access and write access to input and output objects respectively.
Object Instance	Instance of the object. The instance must be unique for each object type and BACnet device, i.e. for each device, there can be only one analog input object 1.
Keyname	This is normally identical to the BACnet object name. Governed by the Desigo Insight naming conventions. Automatically derived from the OPC item name during OPC import, provided the option **Suggest BACnet Object Name and Keyname** is enabled. The key name is interpreted on the Desigo-side as the **User Designation**.

Column	Description
BACnet Object Name	Name of the BACnet object. Governed by the Desigo Insight naming conventions. A suggested name based on the OPC item can be generated automatically during the import process. The object name is interpreted on the Desigo-side as the **Technical Designation**.
Description (Optional)	**Description** of the object. If several mapping rules exist for the same object, it is sufficient to fill in the description in the first row.
BACnet Property (Optional)	Reference to the **BACnet Property** to which the OPC value is to be mapped. If blank, [PrVal] is used. [PrVal] is always mapped from OPC to BACnet for input objects, and from BACnet to OPC for output and value objects.
Destination	For all mapping to properties not related to [PrVal], the direction for the transmission of changes of value must be specified. The following applies: BACnet = Always write value from the OPC side to the **BACnet Property**. OPC = Always write value from **BACnet Property** to OPC item. Twoway = Values can be written in both directions
Force Update	For further information refer to [2)] Flag for change handling: 0 = No change in this row 1 = Row modified Forces SX Open to reset the internal data to the data defined in SX Configurator for the object, e.g. **High Limit**, **Low Limit**, etc. (The object data is usually modified by the user during normal operation, after the initial creation of the SX Open database. **ForceUpdate** resets these values to those in SX Configurator.)
Relinquish Default	Default for [PrVal] in the output object and value object, if no other value is defined. This only makes sense after starting SX Open in the absence of OPC scanning. If blank, the value '0' is generated.

[1)] BACnet Device Ref

As a rule, the data to be integrated via SX Open is assigned to specific device. It is now useful to group the data on the BACnet side in devices, in turn. For this purpose, relevant devices should be defined in the **BACnet Device** sheet, to which the associated (value) objects can be assigned via the **BACnet Device Ref** column. For each BACnet device itself, a BACnet object is generated, of the object type **Device**. The devices are shown with their associated objects in the object tree of a BACnet client.

The rows must be sorted by **BACnet Device Ref**, **BACnet Object Type** and **Object Instance** in that order of priority.

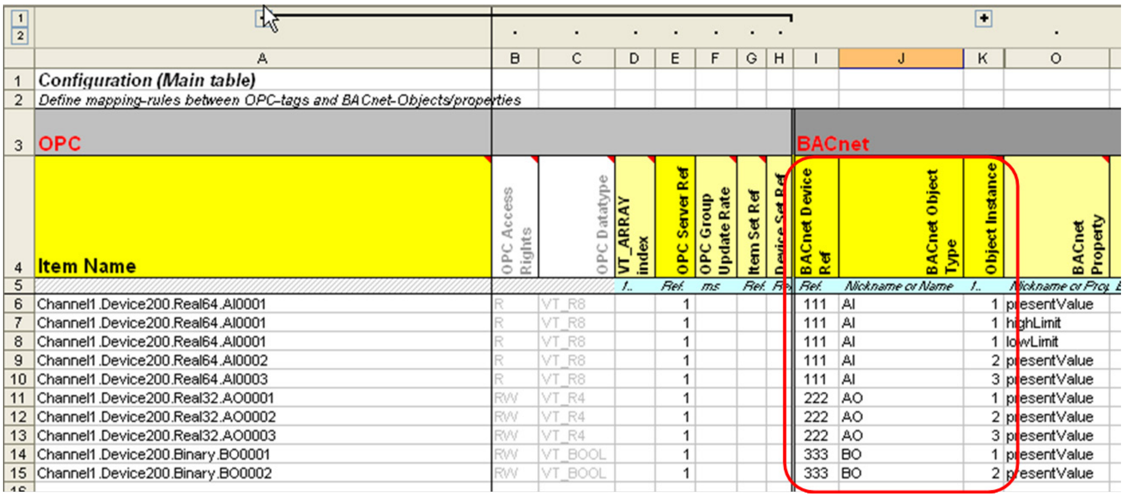


Figure 5-15

[2] Force Update

On start-up SX Open reads the current configuration and compares it with the configuration held in its internal data storage system.

Property values of objects for which the **Force Update** column in the **Configuration** sheet is set to 1 are overwritten with values from csv file. Properties of all other objects are read from internal data storage system.

For a new configuration, the **ForceUpdate** column is set to 1 for all objects. On start-up, therefore, SX Open adopts the complete configuration and generates all objects. Only if a configuration already in use is modified via SX Configurator is it necessary to make sure that the **ForceUpdate** column is set to 1 in all rows containing modified objects. Otherwise, SX Open would ignore these rows when reloading the configuration, and would continue to run with the existing, unmodified configuration.

STOP **Note**

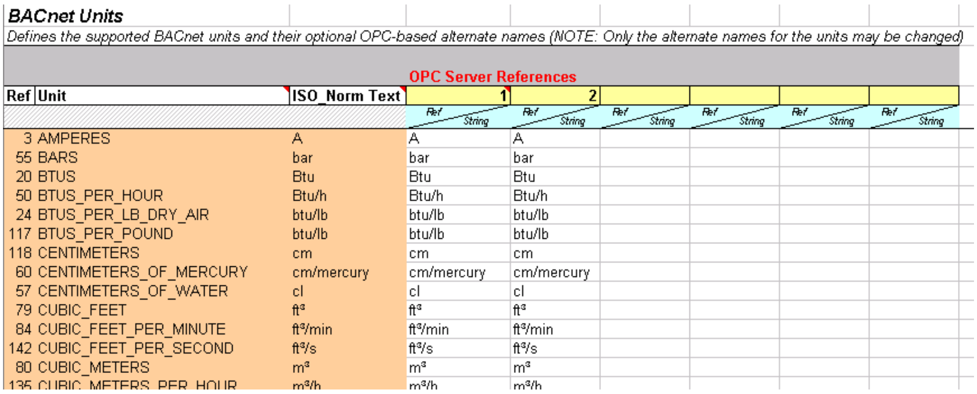
Every successful compilation automatically sets **ForceUpdate** to 0.

5.2.3 BACnet information for analog objects

<table>
<tr><th>Column</th><th>Description</th></tr>
<tr><td>BACnet Units

Important</td><td>For analog objects only: unit for the analog object. This can be the BACnet unit number (e.g. 62), the BACnet unit name (DEGREES_C) or the OPC-server unit string ("°C").

Reference to engineering unit in the Unit sheet:


BACnet Units
Defines the supported BACnet units and their optional OPC-based alternate names (NOTE: Only the alternate names for the units may be changed)
<table>
<tr><th></th><th></th><th></th><th colspan="6">OPC Server References</th></tr>
<tr><th>Ref</th><th>Unit</th><th>ISO_Norm Text</th><th>1</th><th>2</th><th></th><th></th><th></th><th></th></tr>
<tr><th></th><th></th><th></th><th>Ref / String</th><th>Ref / String</th><th>Ref / String</th><th>Ref / String</th><th>Ref / String</th><th>Ref / String</th></tr>
<tr><td>3</td><td>AMPERES</td><td>A</td><td>A</td><td>A</td><td></td><td></td><td></td><td></td></tr>
<tr><td>55</td><td>BARS</td><td>bar</td><td>bar</td><td>bar</td><td></td><td></td><td></td><td></td></tr>
<tr><td>20</td><td>BTUS</td><td>Btu</td><td>Btu</td><td>Btu</td><td></td><td></td><td></td><td></td></tr>
<tr><td>50</td><td>BTUS_PER_HOUR</td><td>Btu/h</td><td>Btu/h</td><td>Btu/h</td><td></td><td></td><td></td><td></td></tr>
<tr><td>24</td><td>BTUS_PER_LB_DRY_AIR</td><td>btu/lb</td><td>btu/lb</td><td>btu/lb</td><td></td><td></td><td></td><td></td></tr>
<tr><td>117</td><td>BTUS_PER_POUND</td><td>btu/lb</td><td>btu/lb</td><td>btu/lb</td><td></td><td></td><td></td><td></td></tr>
<tr><td>118</td><td>CENTIMETERS</td><td>cm</td><td>cm</td><td>cm</td><td></td><td></td><td></td><td></td></tr>
<tr><td>60</td><td>CENTIMETERS_OF_MERCURY</td><td>cm/mercury</td><td>cm/mercury</td><td>cm/mercury</td><td></td><td></td><td></td><td></td></tr>
<tr><td>57</td><td>CENTIMETERS_OF_WATER</td><td>cl</td><td>cl</td><td>cl</td><td></td><td></td><td></td><td></td></tr>
<tr><td>79</td><td>CUBIC_FEET</td><td>ft^3</td><td>ft^3</td><td>ft^3</td><td></td><td></td><td></td><td></td></tr>
<tr><td>84</td><td>CUBIC_FEET_PER_MINUTE</td><td>ft^3/min</td><td>ft^3/min</td><td>ft^3/min</td><td></td><td></td><td></td><td></td></tr>
<tr><td>142</td><td>CUBIC_FEET_PER_SECOND</td><td>ft^3/s</td><td>ft^3/s</td><td>ft^3/s</td><td></td><td></td><td></td><td></td></tr>
<tr><td>80</td><td>CUBIC_METERS</td><td>m^3</td><td>m^3</td><td>m^3</td><td></td><td></td><td></td><td></td></tr>
<tr><td>135</td><td>CUBIC_METERS_PER_HOUR</td><td>m^3/h</td><td>m^3/h</td><td>m^3/h</td><td></td><td></td><td></td><td></td></tr>
</table>
Alternatively, the OPC unit can be used, if shown in the Unit sheet.

For OPC server reference 1, the following can be entered here (or in the Device Set sheet) as the unit for "ampere", for example: A or AMPERES or 3.

The column ISO_Norm Text is used for a text representation when generating EDE files or when calling DIEPT. If this column is empty the Unit column will be used instead.

The OPC Server reference is matched against the Ref column in OPC Server sheet. If more OPC Servers are used in one configuration more columns to Unit sheet must be added.

If the unit string does not exist the Unit sheet, but is shown in the Configuration sheet (or Device Set sheet), an error message appears, e.g.:

<code>**ERROR** Row 46: Invalid unit for OPC Server reference #1: s</code>

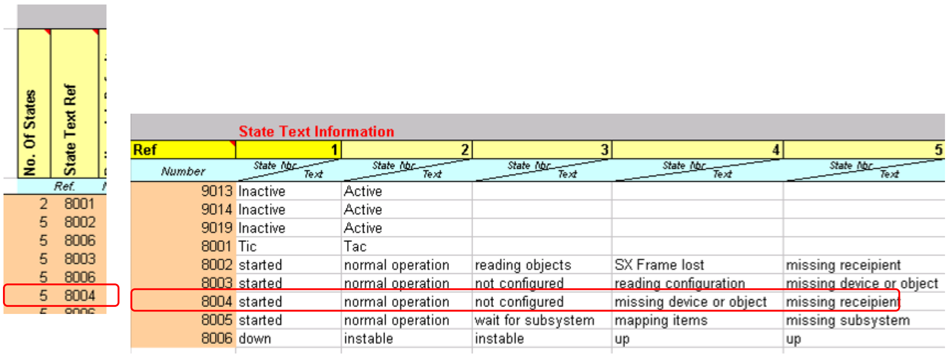
Either add the string to the Unit sheet, or, in the case of the error shown in the example above, change the BACnet Unit in the Configuration sheet (or Device Set sheet) to 73 or SECONDS for all rows displaying "s".</td></tr>
<tr><td>Min Present Value
(optional)</td><td>Min Present Values describes the minimum value for analog objects. For analog-input and analog-value objects receiving a value below Min Present Value will generate a fault event, for analog-output and analog-value objects trying to write a value below Min Present Value will be rejected.</td></tr>
<tr><td>Max Present Value
(optional)</td><td>Max Present Values describes the maximum value for analog objects. For analog-input and analog-value objects receiving a value above Min Present Value will generate a fault event, for analog-output and analog-value objects trying to write a value above Max Present Value will be rejected.</td></tr>
<tr><td>COV increment
(optional)</td><td>COV increment is used to decide which amount a value must be modified before this modification is detected as a change and a COV notification is generated.</td></tr>
<tr><td>Deadband
(optional)</td><td>Deadband assign a value to the BACnet property deadband.</td></tr>
<tr><td>Resolution
(optional)</td><td>Enter a value for resolution. As a common rule of BACnet COV increment must be at least 2*resolution.</td></tr>
<tr><td>Update Rate
(optional, analog-input only)</td><td>Enter a value for update rate.</td></tr>
</table>

5.2.4 BACnet information for binary objects

Column	Description
Polarity (optional)	Polarity can be used to change an OPC item to positive logic, i.e. 0 as passive state and 1 as active state.
Active Text (optional)	Active text assigns a value for present-value 1.
Inactive Text (optional)	Passive text assigns a value for present-value 0.
Min Off Time (optional)	Min Off Time describes how long present-value stays at 0 before a change back to 1 is allowed again.
Min On Time (optional)	Min On Time describes how long present-value stays at 1 before a change back to 0 is allowed again.

5.2.5 BACnet information for multistate objects

<table>
<tr><th>Column</th><th>Description</th></tr>
<tr><td>No. of States
(required)</td><td>SX Open supports multistate-objects with up to 127 states. With No. of States the actual number of used states is assigned.</td></tr>
<tr><td>State Text Ref</td><td>State Text Ref is a number which references to State sheet. In State sheet there is for every possible state a text to describe the meaning of this state.



No. of States and the number of text entries in the appropriate State sheet line must be in sync.</td></tr>
</table>

5.2.6 BACnet alarm info section

BACnet alarm info								
High Limit	Low Limit	Alarm Value(s)	Fault Values	Notification Class	Message Text	Notify Type	Event Enable	Time Delay

Figure 5-16

SX Open supports the alarm system function in a way similar to that used in Desigo. The alarm function in each alarm-generating BACnet object can be configured accordingly.

Notification classes

SX Open uses a sheet **NC** to define a set of notification classes.

Notification Classes

Defines the Notification Classes with their corresponding properties, which are referenced from the Configuration Sheet

NC-ID	Keyname	BACnet Object Nar	Decription	Priority TO_OFFNORMAL	Priority TO_FAULT	Priority TO_NORM.	Ack_Required TO_OFFNORMAL	Ack_Required TO_FAULT	Ack_Required TO_NORMAL	Static Recipient 1	Static Recipient 2	Static Recipient 3
Number	*String*	*String*	*String*									
11	NotifClass11	NotifClass11	Urgent alarm - simple	1	1	5	0	0	0			
12	NotifClass12	NotifClass12	Urgent alarm - basic	1	1	5	1	1	0			
13	NotifClass13	NotifClass13	Urgent alarm - extended	1	1	5	1	1	1			
21	NotifClass21	NotifClass21	High prio alarm - simple	2	2	6	0	0	0			
22	NotifClass22	NotifClass22	High prio alarm - basic	2	2	6	1	1	0			
23	NotifClass23	NotifClass23	High prio alarm - extended	2	2	6	1	1	1			
31	NotifClass31	NotifClass31	Normal alarm - simple	3	3	7	0	0	0			
32	NotifClass32	NotifClass32	Normal alarm - basic	3	3	7	1	1	0			

Figure 5-17

SX Configurator is delivered with a set of predefined notification classes which are used in Desigo, too.

Column	Description
NC-Id	NC-Id is the instance-number of the corresponding notification-class BACnet object. The predefined notification-classes are assigned instance numbers according to Desigo.
Keyname	In this column the keyname is entered
NC-Name	In this column the object-name is entered
Description	**Description** of the notification-class object.
Priority TO_OFFNORMAL	Priority of a TO_OFFNORMAL alarm or event.
Priority TO_FAULT	Priority of a TO_FAULT alarm or event.
Priority TO_NORMAL	Priority of a TO_NORMAL alarm or event.
Ack_Required TO_OFFNORMAL	0 = no acknowledged required 1 = TO_OFFNORMAL transition state has to be acknowledged
Ack_Required TO_FAULT	0 = no acknowledged required 1 = TO_FAULT transition state has to be acknowledged
Ack_Required TO_NORMAL	0 = no acknowledged required 1 = TO_NORMAL transition state has to be acknowledged
Static Recipient 1 **Static Recipient 2** **Static Recipient 3**	Persistent alarm recipients: For each notification class object, up to three persistent (fixed) alarm recipients can be configured via the associated Device ID. The advantage of persistent alarm recipients is that they can receive alarms immediately after a restart, without the need to log into SX.

To assign a notification class to a BACnet object the value in column Notification Class of **Configurator** sheet must be filled accordingly.

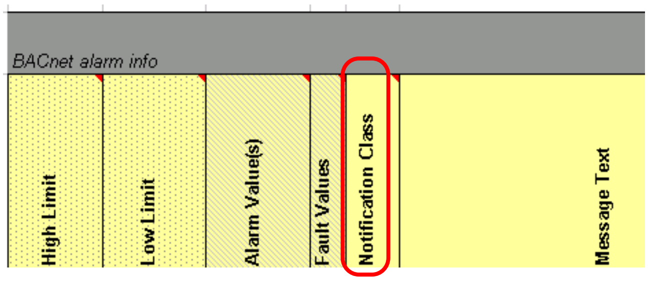


Figure 5-18

To specify the circumstances an event transition shall occur, alarm values (for binary and multistate objects), alarm limits (for analog objects) and fault values (for multistate objects) must be assigned to each of the BACnet objects required to generate alarms or events.

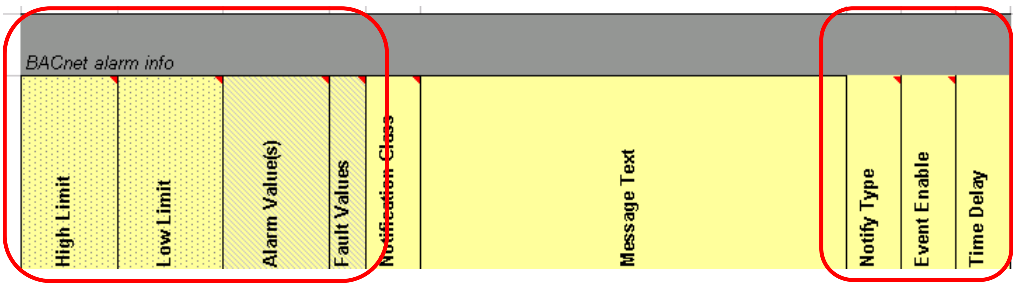


Figure 5-19

Column	Description
High limit	For analog objects only: If an analog object is required to generate a BACnet alarm, a value must be entered in either the High Limit or the Low Limit column (or both). SX generates an OFFNORMAL alarm if [PrVal] violates the limit. For further information refer to [1)] Blank = No alarm Value = High alarm limit x = Alarm limit from OPC item
Low limit	For analog objects only: SX generates an OFFNORMAL alarm if [PrVal] drops below/above the limits. For further information refer to [1)] Blank = No alarm Value = Low limit alarm x = Alarm limit from OPC item
Alarm Value(s)	For binary and multistate input objects and value objects only: If a multistate object is required to generate a BACnet alarm, the Alarm Value(s) column must contain a list of up to 127 comma-separated valid state values corresponding to alarm states. SX generates an OFFNORMAL alarm whenever the object matches one of the alarm values. Binary objects = 0 or 1 Multistate objects = one or more values from 1…127, separated by commas (e.g. 2,5,11 etc.)
Fault Values	For multistate input objects and value objects only: If a multistate object is to generate a BACnet alarm, this column must contain a list of up to 127 comma-separated valid state values corresponding to alarm states. SX generates a FAULT alarm if [PrVal] of the object corresponds to one of the fault values. This column must contain **numbers** only, and not text. empty = No faule value assigned One or more values from 0…127, separated with commas (e.g. 2,5,11 etc.)
Message Text (Optional)	An optional alarm message text can be configured separately for each object, for display in the Log Viewer and Alarm Viewer applications of Desigo Insight.
Notify Type (optional)	Enter as applicable: 0 = send state-transition as an alarm 1 = send state-transition as an event
Event Enable (optional)	Enter as applicable (addition of 3 event enable bits): 1 = to-offnormal enabled 2 = to-fault enabled 4 = to-normal enabled
Time Delay (optional)	Time Delay is used by SX Open to decide how long a TO_OFFNORMAL state must be present before a state-transition is reported.

[1] High/Low Limit

For analog objects, the alarm function is activated via these columns. If the columns are left blank, no alarm is initiated.

Alarm limits can be defined in the following two ways:

- Directly, by specifying a physical value in the same format as the [PrVal] of the object.
- Indirectly, via the value from another OPC item.
  In this case, an "x" must be entered instead of the value.

In addition, in compliance with another mapping rule, the OPC item which defines the alarm value must be assigned to the high/low alarm limit of the same BACnet object.

Example

*OPC Item	BACnet Object=	BACnet Property=	High Limit=
Item_1	ANALOG_IN_1	Present_Value [IPrVal]	**x**
Item_2	ANALOG_IN_1	**High_Limit**	

For further information on the alarm function in SX Open, refer to the Engineering Guideline [1] and the Desigo "Technical principles" manual, [5].

5.2.7 BACnet trend info section

SX Open supports the trend logging system function in a way similar to that in Desigo PX. Trending is implemented as a relatively autonomous function, and operates as an independent BACnet client of SX Open (SX trend server with its own data maintenance for trend log objects).

As an option, SX can log trends for the present values of the SX BACnet objects. For this purpose, a trend log object must be assigned to the BACnet object at the configuration stage.

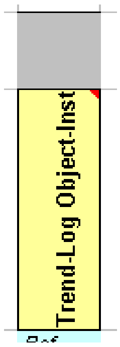


Figure 5-20

Column	Description
Trend-Log object instance	Enter the instance number of the trend log BACnet object. The instance number must be unique and must occur once only in the file. When compiling SX configuration data a separate sheet **Trend** will be filled with trendlog objects based on instance numbers used in **Configuration** sheet.

STOP **Note**

The generated **Trend** sheet can be modified after a first compilation and with the next compilation the modified values will be used.
To assign a unique keyname, object name and device instance BACnet device sheet must be adjusted.

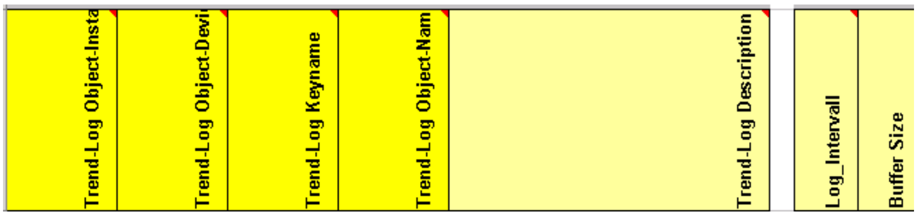


Figure 5-21

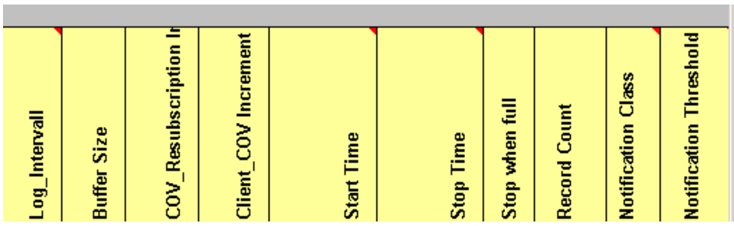


Figure 5-22

Column	Description
Trend-Log Object-Instance	The instance number is referenced from **Configuration** sheet and may not be changed.
Recommendation	When modifying the configuration at a later date, take care not to change any instance numbers already allocated to trend log objects. The instance number allocated to the "trended" BACnet object must also be retained.
Important	Failure to observe these rules can result in the loss of trend data.
Trend-Log Object Device ID	The instance number of the device where the BACnet object to monitor is located. This entry may not be changed, too.
Trend-Log Keyname	Keyname of trendlog object.
Trend-Log Object Name	The name of the trendlog object. The name is generated to the rule "TRD" + instance number, in the form XXXX, e.g. TRD0045", but can be changed afterwards.
Trend-Log Description **Important**	"ObjectName of the assigned object" supplemented with buffer size and log interval e.g. "OutdoorTemperature: 5000 records, unconfirmed COV" Once a description is generated by compilation it will not be changed afterwards automatically. If e.g. buffer size is changed and the change shall be reflected in **Trend** sheet, trendlog description must be manually deleted.

Column	Description
Log interval	Describes how to get the monitored value: 0 = use confirmed COV notification -1 = use unconfirmed COV notification >0 = poll monitored object and use supplied value as poll interval
Buffer size	Size of the temporary trend buffer for each object. The amount of space required for the trend log objects in the working memory of the SX PC depends on their number and on the buffer size, and is in direct competition with other applications, e.g. other SX components. The buffer size should therefore be defined so as to achieve the optimum relationship between data logging (LogInterval) and storage in Desigo Insight.
Recommendation	Desigo generally limits the size of the trend log object buffer to a maximum of 5000 entries. For SX Open there is also no point in generating buffers that are larger than this.
COV resubscripiton interval	Describes the time between COV resubscription of a monitored object. Currently only a COV resubscription of 0 (unlimited lifetime) is supported.
Client COV Increment	Describes the COV increment between two values received or read before storing a new value in trendlog buffer.
Start time	When to start logging of data.
Stop time	When to stop logging of data.
Stop when full	Describes the behavior, if log buffer is full: 0 = overwrite the oldest entries in log buffer 1 = stop logging further data
Record count	This value may not be modified, it is used if an upload of trendlog objects occurs
Notification class	The value of the first active row will be used as notification class for *all* trendlog objects. If different values are entered in different rows all values will be adjusted to the value of the first active row.
Notification threshold	The amount of data in trendlog buffer before an event will be sent to all registered recipients. If the warning limit is reached (BACnet property "Notification Threshold"), the complete memory content is transferred to a BACnet client for evaluation and archiving. It is advisable to set the warning limit at 40 to 60%.

Trend log file archive

The trended values can also be logged in text form in a file. This can be useful in the commissioning phase, for example, when a trend upload to a BACnet client is not yet possible.

For more information on the trend function in SX Configurator, refer to the Engineering Guidelines [1].

5.2.8 BACnet schedule object section

As an option, SX can change a BACnet object value based on a schedule. For this purpose, a schedule object must be assigned to the BACnet object at the configuration stage.

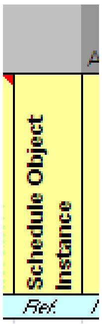


Figure 5-23

Column	Description
Schedule Object Instance	Enter the instance number of the scheduleg BACnet object. The instance number must be unique and must occur only once in the file. When compiling SX configuration data a separate sheet **Schedule** will be filled with schedule objects based on instance numbers used in **Configuration** sheet.

The generated **Schedule** sheet can be modified after a first compilation and with the next compilation the modified values will be used.

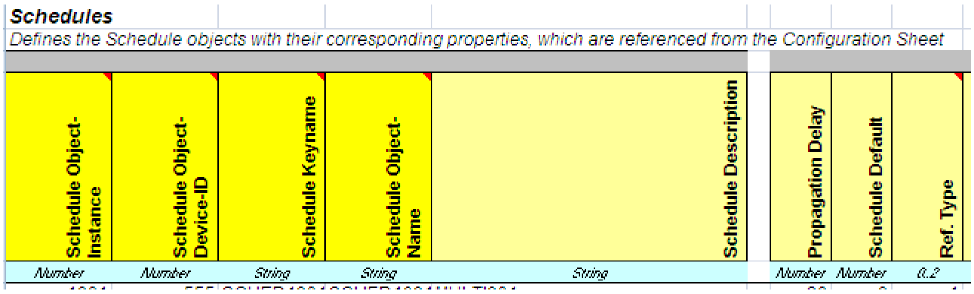


Figure 5-24

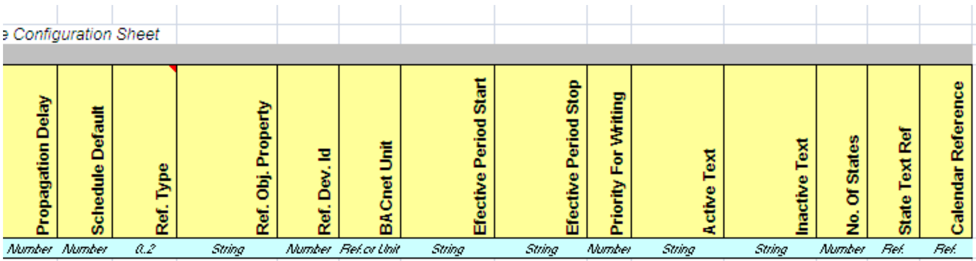


Figure 5-25

Column	Description
Schedule Object Instance	The instance number is referenced from **Configuration** sheet and may not be changed.
Recommendation	When modifying the configuration at a later date, take care not to change any instance numbers already allocated to schedule objects. The instance number allocated to the "scheduled" BACnet object must also be retained.
Important	Failure to observe these rules can result in unexpected schedule results..

Column	Description
Schedule Object Device ID	The instance number of the device where the BACnet object to schedule is located. This entry may not be changed, too.
Schedule Keyname	Keyname of schedule object.
Schedule Object Name	The name of the schedule object. The name is generated to the rule "SCHED" + instance number, in the form XXXX, e.g. SCHED0023", but can be changed afterwards.
Schedule Description **Important**	ObjectName of the assigned object Once a description is generated by compilation it will not be changed afterwards automatically but can always be changed manually
Propagation Delay	Propagation delay for propagatin fault events, see also Configuration sheet
Schedule Default	The default value to write to controlled object when no schedule rule is active
Ref. Type	The type of object reference. Currently only the value 1 is supported.
Ref. Obj. Property	The property of the controlled object which shall be written.
Ref. Dev. Id **Important**	The device instance the controlled object is located. In current implementation the schedule object and the controlled object must be located on the same device.
BACnet Unit	If controlled object is of type analog BACnet unit will be copied from Configuration sheet
Effective Period Start	Schedule is active only if current time if greater than Effective Period Start
Effective Period End	Schedule is active only if current time if less than Effective Period End
Priority for Writing	The priority output from schedule object shall be written to controlled object.
Active Text	If controlled object is of type binary active text will be copied from Configuration sheet
Inactive Text	If controlled object is of type binary inactive text will be copied from Configuration sheet
Number of States	If controlled object is of type multistate number of states will be copied from Configuration sheet
State Text Ref.	If controlled object is of type multistate state text reference will be copied from Configuration sheet
Calendar Reference (optional)	An instance number of a calendar object, If present, reference will be inserted in first exception schedule.

Important

Once BACnet unit, active and inactve text, number of states and state text ref. are copied from Configuration sheet they will not be changed afterwards automatically. Changes in configuration must be copied manually to Schedule sheet.

To create a BACnet calendar object a line has to be inserted in configuration sheet. For calendar objects there is no need for an OPC section in Configuration sheet.

Important

Schedule and calendar objects are created by SX Configurator as container only. Except a calendar reference in first exception schedule there will be no schedule or calendar data inserted by SX Configurator. To insert this data a BACnet client, e.g. Desigo Insight, is needed.

5.2.9 Section alarm suppression

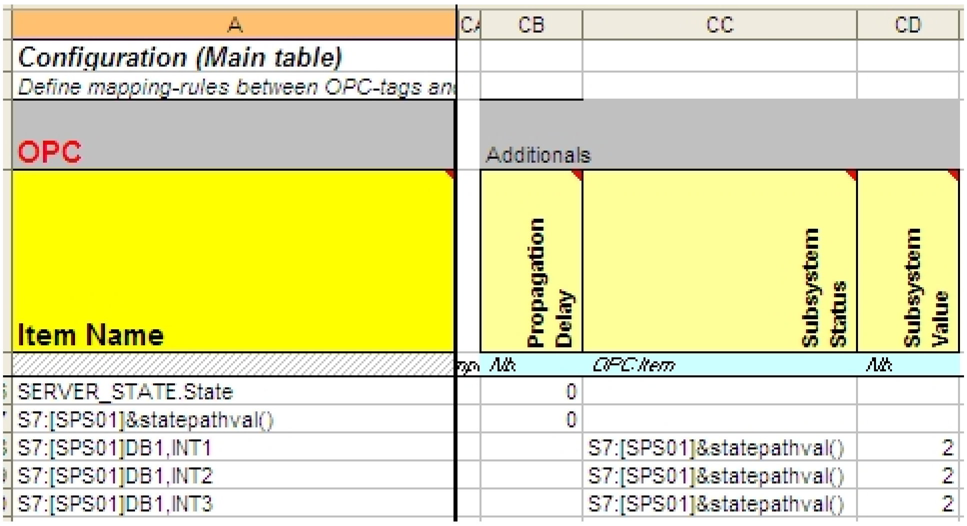


Figure 5-26

Column	Description
Propagation Delay	A delay time before a BAD OPC quality will be delivered as a BACnet fault state. Default value is 30 seconds.
Subsystem Status STOP	An OPC item which describes the communication state to a subsystem. Subsystem status is only useful if an OPC server can distinguish different devices and every device is modeled as a BACnet device of its own.
Subsystem Value	An item value of Subsystem Status, which represents a working connection.

STOP **Note**

Alarm suppression is only useful and does only work if default mapping is enabled in cover sheet (cf. 3.3).

Alarm suppression disables generating of BAD quality status values for OPC items if subsystem-status is not equal to subsystem-value. BAD quality values caused by a communication failure will be suppressed. By delaying a BAD quality propagation for some time there is an independence of update sequence received from OPC server. In some cases BAD quality items are signaled before signaling a broken communication. This leads temporarily to a BAD quality which will go back to GOOD if the communication failure is received.

STOP **Trend**

For objects being monitored by SX Trend Server there is no alarm suppression. If default mapping is enabled reliability of these objects will follow OPC quality independent of connection status. Reliability is part of associated trendlog object logbuffer entry and may not be suppressed in SX BACnet Server. It is recommen-ded to set NotifyType to Event for these objects.

Example

Figure 5-26 shows some parts of a SIMATIC.NET OPC Server configuration. This server provides for every connection an item "&statepathval()" which describes the current state of subsystem communication. A working communication is coded as 2, all other values are used for different types of dead connections.

In the example above a BAD quality of item "S7:[SPS01]DB1,INT1" will be suppressed, if item "S7:[SPS01]&statepathval()" signals a broken connection, i.e. if item value is not equal to 2.

5.2.10 Function info section

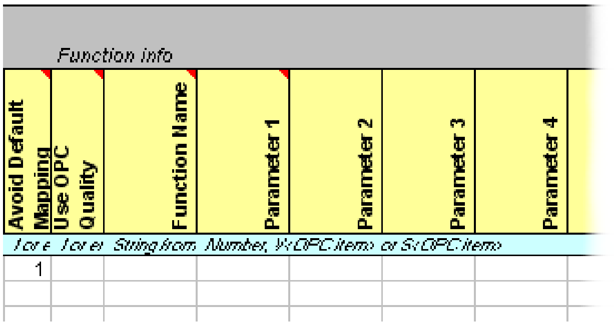


Figure 5-27

Column	Description
Avoid Default Mapping	Disables the default mapping for alarms individually. (Refer also to the information on **Default Mapping** in section 5.1.3.)
Uses OPC Quality	For default mapping of an OPC item to a BACnet property, the OPC item value is used. It may on occasion be necessary to map the **OPC Item Quality**, instead of its value, as the input variable for a function block. In this case, enter a "Y" (or other letter) in this column. This is only possible where the OPC Quality is used for mapping in direction **to** the BAC property. Blank or <>1 = Use item value 1 = Use item status
Function Name	Function name from the Functions sheet, section **5.11**. If bidirectional mapping, the inverse function must be defined, too. The inverse function is called with the same parameter set, the only difference is input value (OPC item or BACnet property)
Parameter 1..40	Function parameters that are restricted to **numbers**, **OPC Item references**, **BACnet properties** and **Hash-Values** for VT_BSTR items. To distinguish different entries the following syntax is used: `V<opc_item>` = value of an OPC item `S<opc_item>` = state of an OPC item `F<objtype:devinst:inst:prop>` = value of an object property[1)] `H<string>` = hash value of string[2)]

[1)] `F<...>` The properties used for this kind of function parameter are properties of SX Frame and therefore an extension of standard BACnet properties.

`objtype`	is a valid object type from **Objects** sheet.
`devinst`	is the instance number of a device object which "hosts” the referenced object.
`inst`	is the object instance of the referenced object itself.
`prop`	is a property from **Properties** sheet.

2) H<...>

To support VT_BSTR items e.g. for signaling a state, these strings must be translated in a unique number which can be handled by SX Open. These strings have to be mapped to hash values in SX Configurator (see **Hash VT_BSTR** sheet). The same hashing function will be used in SX Open as soon as string is received from OPC Client. By defining a mapping function which knows about the different hash values a mapping of strings to a discrete set of numbers is possible.

Important

For direction OPC mapping function which is used to map a BACnet property to an OPC item must be entered. For direction BACnet or Twoway mapping function which is used to map an OPC item to a BACnet property must be entered

For more information on the functions for signal and function mapping refer to the Engineering Guideline [1].

5.2.11 Setting default values

Ref	Property name	Nickname	Default
		String	
1	ackRequired		
0	ackedTransitions		
4	activeText	TXTACT	On
6	alarmValue		
7	alarmValues		
22	covIncrement	COVI	0.1
25	deadband	DBAND	1
28	description		
35	eventEnable	EVTEN	7
130	eventTimeStamps		
39	faultValues		
40	feedbackValue	FbVal	
45	highLimit		3.40E+38
46	inactiveText	TXTINACT	Off
48	instanceOf		
52	limitEnable		

Figure 5-28

To reduce the amount of work in entering values in **Configuration** sheet, default values will be used whenever a field has been left blank. The default value can be assigned by entering a value in column default of **Properties** sheet. If this field is left blank, too, a build in default will be used.

5.3 Item Set sheet

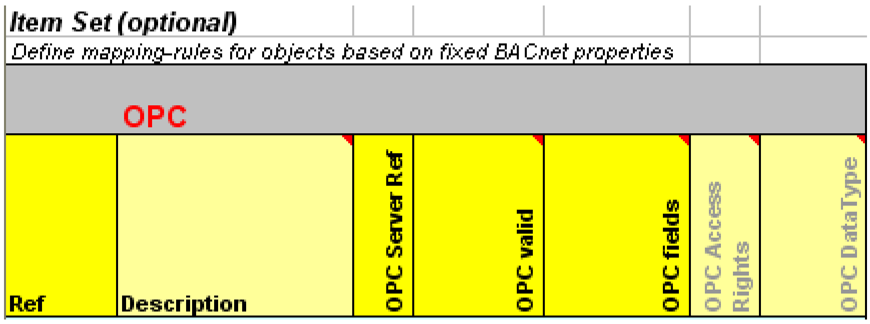


Figure 5-29

- At least one entry must have [PrVal] as a property and must also define the BACnet object type.
- Each entry is mapped with a BACnet property, which means that if not [PrVal], then the **Output Destination** must be defined. .

Defining Item Sets

This sheet allows the separate definition of recurring substructures when mapping OPC items to BACnet properties. If a given structure recurs, all that is needed is a simple reference in the **Configuration** sheet to the actual structure in the **Item Set** sheet. One row in the **Configuration** sheet containing the basic mapping is enough to define the mapping relationships via an item set.

Example

A set may consist of three items, each of which represents a main value, an upper limit (Hili/High_Limit) and a lower limit (Loli/Low_Limit). Within the **ItemSet** sheet, the rule for mapping these three items to BACnet properties is defined just once. In the **Configuration** sheet, the basic item is simply assigned, on one row, to the BAnet object. The Item Set itself then also has to be referenced. In the compile process, this "nested" relationship is unraveled and downloaded to SX Open as three rows.

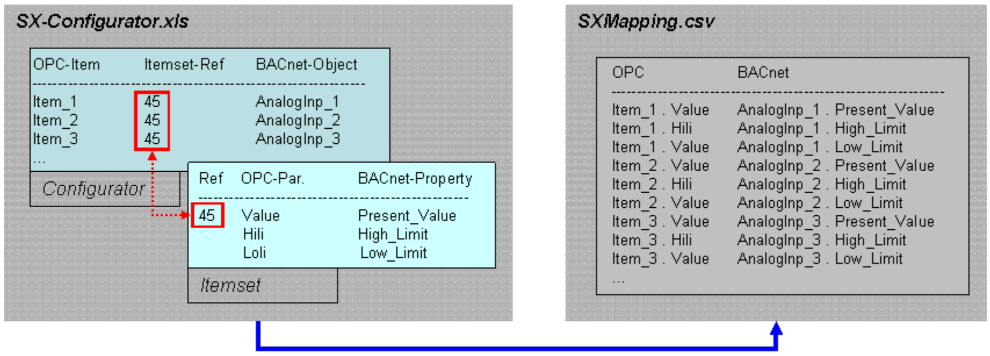


Figure 5-30

The **Item Set** sheet contains a subset of the columns in the **Configuration** sheet. The table below only explains those columns not already dealt with under **Configuration** sheet.

Column	Description
Ref	Unique set number used as a reference in the **Configuration** sheet. The **Ref**erence is a sequence number in two parts: **main-item.sub-item**; all definitions related to the same object must have the same **main-item** reference; The reference number is divided into two parts, x.y: x represents the set number to which reference is made in the **Configuration** sheet. y is the numbering within the item set.
OPC Server Ref	The **OPC Server Ref** must be known. This means that a **full** import has already taken place, or that the information has been entered manually in the **OPC Server** sheet.
OPC valid	This is the part of an item name which is both fixed and identical. **OPC valid** contains a **validation string**, i.e. the OPC item to be used for this entry must contain this string. The members of a set are identified by this fixed component of the name. For further information see [1)]
OPC fields	This defines the variable component of the item name, which is more or less equivalent to a parameter name. The **OPC fields** contain the fields to the right, which refer to specific BACnet properties and must start with a dot or period (**.**). The variable component must always begin with a leading dot or period, "**.**". For further information see [1)]

[1)] OPC valid/OPC fields

OPC item sets are defined by name. Item sets can be defined only if the names of the OPC items are regular in structure.

The structure can be said to be "regular" if it complies with the following:

- The item names must follow a hierarchical structure, e.g.
  System.Device.Data point.Parameter
  and
- the name fields must be separated by a dot or period "**.**".

In principle, then, the **OPC valid** column defines the fixed, common component of an item name within the set (equivalent to the data point field or the common object address in the item name), and the **OPC fields** column defines the variable component of an item name within the set (equivalent to the parameter field in the item name). In conjunction with the basic mapping rule in the **Configuration** sheet, SX Configurator can resolve the item set when compiling the configuration.

Example

The following OPC data is imported into SX Configurator with a **Full Import**.

```
…
RSAcc.QC0101-NRUD (1).VC01-FC20.RSUAO1.Value
RSAcc.QC0101-NRUD (1).VC01-FC20.RSUAO1.Status
RSAcc.QC0101-NRUD (1).VC02-FC22.RSUAO2.Value
RSAcc.QC0101-NRUD (1).VC02-FC22.RSUAO2.Status
RSAcc.QC0101-NRUD (1).VC03-FC18.RSUAO3.Value
RSAcc.QC0101-NRUD (1).VC03-FC18.RSUAO3.Status
…
```

It can be seen that a value and a status must be mapped for each identically addressed pair.

To simplify the configuration process, it also possible to define just one **Item Set** with:

Ref =	**OPC valid =**	**OPC fields =**
1	**RSUA0**	**.Value**
1.1	**RSUA0**	**.Status**

The basic links in the **Configuration** sheet are created automatically after the item set definition process, when an OPC data import is next started. This time, however, the import type is **Number of Fields**. SX Configurator can then identify the sets automatically from the set information, and only uses the reduced basic links in the **Configuration** sheet (see also section 5.2).

5.4 Device Set sheet

Device Set (optional)								
Define mapping-rules for objects based on a set of BACnet objects								
OPC								
Ref	Description	Info	OPC Valid	Item	OPC Access Rights	OPC DataType	VT_ARRAY index	OPC Group Update Rate
Number	*String*	*See comment*	*String*	*Partial OPC Item String*			*/..*	*ms*

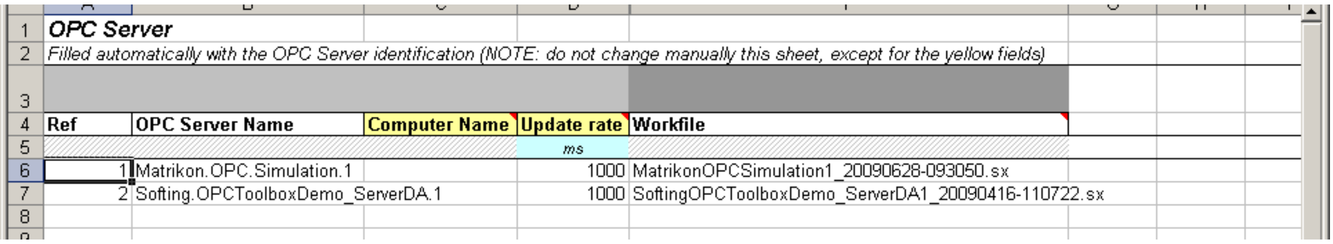
Figure 5-31

The **Device Set** sheet and the **Item Set** sheet work in similar ways. A device set, however, allows you to group together all items which can be assigned to a physical device and which always have to be mapped to BACnet objects and BACnet properties in the same way..

Then you only need to configure one row in the **Configuration** sheet with the name and a start number for the instance numbering of the BACnet objects.

This reduces and simplifies the configuration time required for these kinds of integration.

5.5 OPC Server sheet



OPC Server

Filled automatically with the OPC Server identification (NOTE: do not change manually this sheet, except for the yellow fields)

Ref	OPC Server Name	Computer Name	Update rate	Workfile
			ms	
1	Matrikon.OPC.Simulation.1		1000	MatrikonOPCSimulation1_20090628-093050.sx
2	Softing.OPCToolboxDemo_ServerDA.1		1000	SoftingOPCToolboxDemo_ServerDA1_20090416-110722.sx

Figure 5-32

This sheet is used to store OPC server data, such as name, location and update rate.

Normally, if the OPC item data is imported directly from a specific OPC server, the sheet is filled in automatically. However, you can also add to or correct the sheet manually.

Column	Description
Ref	Unique number of the OPC server, used as a reference in the **Configuration** sheet.
OPC Server Name	Name of the OPC server, e.g. "Matrikon.OPC.Simulation.1"
Computer Name	Network name of the computer on which the OPC server is installed. The name is mandatory if SX Open and the OPC server are installed on different computers.
Update rate	The default rate in milliseconds at which the data in the OPC server is normally scanned. For further information see [1]
Workfile	The OPC server data imported online is always saved in a separate file. This means that there is no need for an online connection to the server if a further import is required. The workfile is saved under ***<OPCServerName>_<Date-Time>*.sx**

[1] Update Rate

Depending on the number of items and changes of value, it is advisable to set a suitably conservative value here, to avoid impairing the performance of SX Open.

5.6 BACnet Device sheet

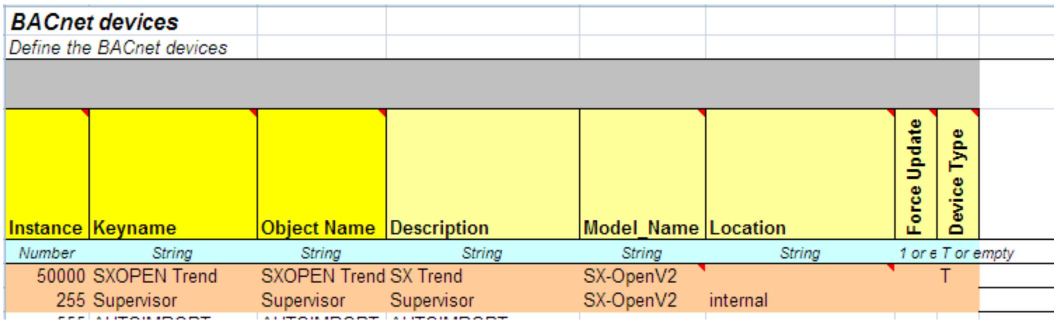


Figure 5-33

All the information about the "virtual" BACnet devices is saved here.
It is best to assign the BACnet objects according to the subsystem or device to be integrated, to achieve a clear, hierarchical object structure on the BACnet side.

Column	Description
Instance	Instance of the BACnet device object. The instance must be unique. It is also used as a reference in the **Configuration** sheet.
Keyname	This is normally identical to the BACnet object name. The key name is interpreted on the Desigo side as the **User Designation**.
Object Name	Name of the BACnet object. This can be derived automatically from the OPC name by SX Configurator, provided this option was enabled before the OPC import. The object name is interpreted on the Desigo side as the **Technical Designation**.
Description	**Description** of the object.
Model Name	Individual description of the SX application, displayed in the device object property **Model_Name**.
Location	Individual description of the location, displayed in the device object property **Location**.
Force Update	See also Configuration Sheet
Device Type	Empty = a BACnet device usable in SX BACnet Server (Configuration Sheet) T = Device for SX Trend Server

5.7 Units sheet

BACnet Units

Defines the supported BACnet units and their optional OPC-based alternate names (NOTE: Only

Ref	Unit	ISO_Norm Text	OPC Server References 1 Ref/String	2 Ref/String	Ref
3	AMPERES	A	A	A	
55	BARS	bar	bar	bar	
20	BTUS	Btu	Btu	Btu	
50	BTUS_PER_HOUR	Btu/h	Btu/h	Btu/h	
24	BTUS_PER_LB_DRY_AIR	btu/lb	btu/lb	btu/lb	
117	BTUS_PER_POUND	btu/lb	btu/lb	btu/lb	
118	CENTIMETERS	cm	cm	cm	
60	CENTIMETERS_OF_MERCURY	cm/mercury	cm/mercury	cm/mercury	
57	CENTIMETERS_OF_WATER	cl	cl	cl	
79	CUBIC_FEET	ft³	ft³	ft³	
84	CUBIC_FEET_PER_MINUTE	ft³/min	ft³/min	ft³/min	
142	CUBIC_FEET_PER_SECOND	ft³/s	ft³/s	ft³/s	
80	CUBIC_METERS	m³	m³	m³	
135	CUBIC_METERS_PER_HOUR	m³/h	m³/h	m³/h	
85	CUBIC_METERS_PER_SECOND	m³/s	m³/s	m³/s	
105	CURRENCY1	MWh	MWh	MWh	

Figure 5-34

This sheet lists all the units defined in the BACnet standard, together with their respective reference numbers.

For each OPC server, you can use the **Unit** sheet to define your own unit designations or the existing ones, for use in the **Configuration** sheet. This can make it easier to enter **Configuration** sheet data in cases where the data needs to be copied and pasted from OPC into Excel.

Column	Description
Ref	BACnet reference number for the unit.
Unit	Unit designation as defined in the BACnet standard.
ISO_Norm Text	This text – if present – will be used when generating EDE files and when using the integrated DIEPT.
Ref/String	Reference to the OPC server in the **OPC server** sheet and the associated unit designation as used on the OPC side.

5.8 State sheet

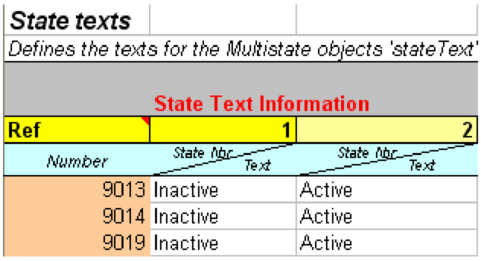


Figure 5-35

This sheet can be used to associate text with the numerical states of multistate objects.

Column	Description
Ref	BACnet reference number for the state
State Nbr/Text	State number and associated text. SX Open supports up to 127 state texts.

Note that in contrast to OPC on the BACnet side, the state 0 is not supported. In such cases, the BACnet state must be made compatible by use of an "SxAdd 1" function block.

If a state is not defined, the entry <undefined> must appear in the field concerned. Blank text fields are not permissible.

5.9 Objects sheet

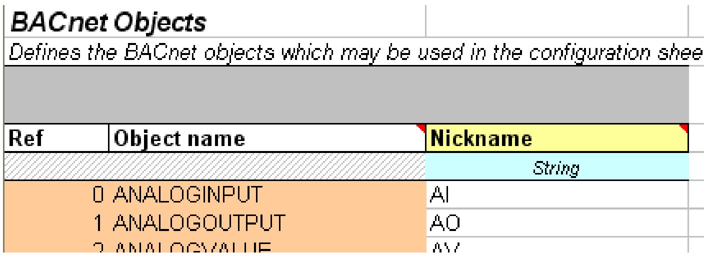


Figure 5-36

This sheet lists the BACnet object types supported by SX Open for mapping the OPC data. To avoid having to enter the object type in full in the **Configuration** sheet, so-called "Nicknames" can be defined. For example "ANALOGINPUT" can be abbreviated to "AI".

Column	Description
Ref	BACnet reference number for the object type
Object Name	Object type as defined in the BACnet standard.
Nickname	Unique abbreviation for use in the **Configuration** or **Standard Properties** sheets.

5.10 Properties sheet

BACnet Properties

Defines the BACnet properties which may be used in the configuration sheets and the

Ref	Property name	Nickname	Default
		String	
1	ackRequired		
0	ackedTransitions		
4	activeText	TXTACT	On
6	alarmValue		
7	alarmValues		
22	covIncrement	COVI	0.1
25	deadband	DBAND	1
28	description		
35	eventEnable	EVTEN	7
130	eventTimeStamps		
39	faultValues		
40	feedbackValue	FbVal	
45	highLimit		3.40E+38
46	inactiveText	TXTINACT	Off
48	instanceOf		

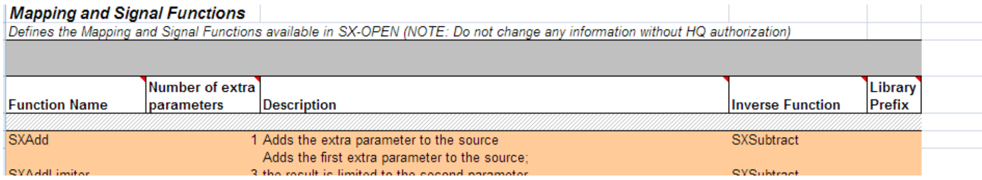
Figure 5-37

This sheet lists the BACnet properties supported by SX Open for mapping the OPC data. To avoid having to enter the object type in full in the **Configuration** sheet, so-called "Nicknames" (abbreviations) can be defined. For example the property "HighLimit" can be abbreviated to "HiLm" and this can be used as a reference.

Each parameter is defined as a tuple (**Property**, **Value** and **Default**).

Column	Description
Ref	BACnet reference number for the property
Property Name	Property name as defined in the BACnet standard.
Nickname	Unique abbreviation for use in the **Configuration** sheet.
Default	If not empty, this value will be used as a default value for his property.

5.11 Functions sheet



Mapping and Signal Functions

Defines the Mapping and Signal Functions available in SX-OPEN (NOTE: Do not change any information without HQ authorization)

Function Name	Number of extra parameters	Description	Inverse Function	Library Prefix
SXAdd	1	Adds the extra parameter to the source	SXSubtract	
		Adds the first extra parameter to the source;		

Figure 5-38

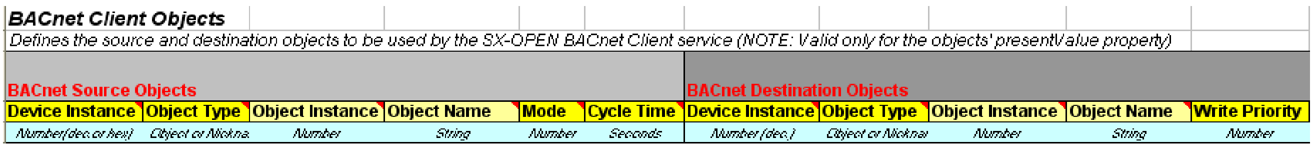
The **Function** sheet is for entry of the predefined functions available via the function block library, **PRJLib.dll**.

Only the functions that are actually needed in the main **Configuration** sheet should be described.

Column	Description
Function Name	Function name
Number of extra Parameters	Number of optional extra parameters
Description	Descriptive text
Inverse Function	Inverse function, for use in cases where the function is used for signal mapping.
Library Prefix	Empty for functions located in PRJLib and EXT for mapping functions located in a private mapping library.

For more information on the functions for signal and function mapping, refer to the SX Open Engineering Guidelines [1].

5.12 BACnet Client sheet



BACnet Client Objects

Defines the source and destination objects to be used by the SX-OPEN BACnet Client service (NOTE: Valid only for the objects' presentValue property)

BACnet Source Objects						BACnet Destination Objects				
Device Instance	Object Type	Object Instance	Object Name	Mode	Cycle Time	Device Instance	Object Type	Object Instance	Object Name	Write Priority
Number(dec or hex)	Object or Nickna	Number	String	Number	Seconds	Number (dec.)	Object or Nicknai	Number	String	Number

Figure 5-39

SX Open normally operates as a BACnet server for the purpose of reading data from third-party devices or systems for various uses in Desigo..

In the reverse process, however, where data is to be transmitted from Desigo to third-party devices or systems, SX Open must also be able to act as a client on the BACnet side. In SX Open this facility is also included as an autonomous function, allowing the interconnection of various BACnet objects. This means, for example, that a connection (and hence the mapping of a value) can be configured between a Desigo PX BACnet object and an SX BACnet object. As soon as there is a change of value in the PX object, the value is transmitted to the connected SX object. As this, in turn, is connected to OPC, the value is also output on the OPC side.

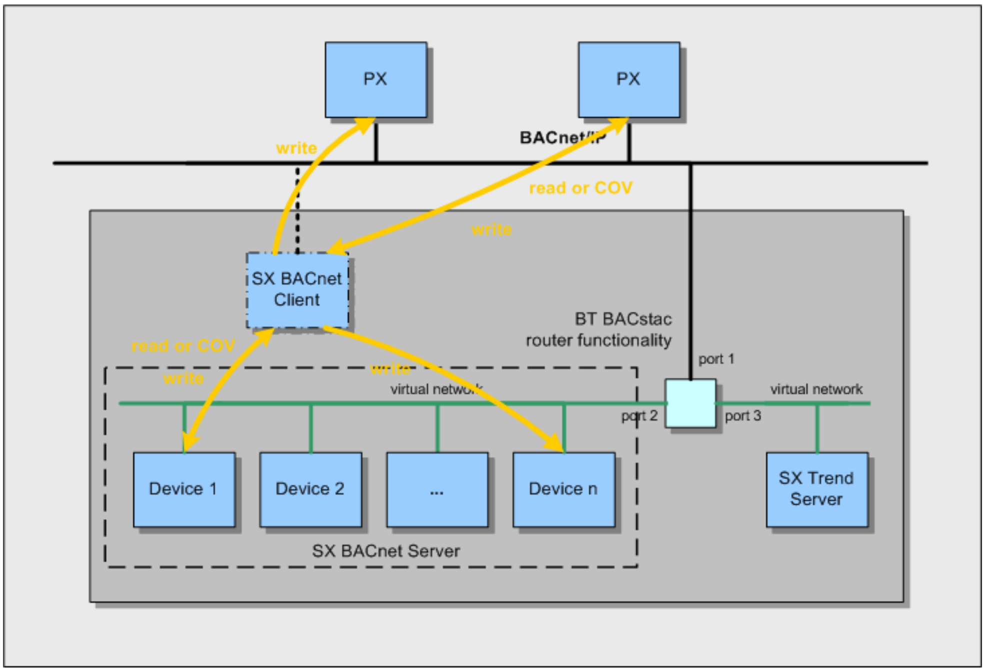


Figure 5-40

Configuring a BACnet client

The client connections, too, can be configured in SX Configurator.

The sheet has two sections, for **Source** and **Destination**. The **Source** section contains the address of the object which is the source of the value (e.g. a DMAP object from PX), and the **Destination** section contains the address of the object to which the value is to be transmitted (e.g. an SX-BACnet object). The full address of an object comprises the following:

- Device instance
- Object type
- Object instance
- Object name

SX can either subscribe to changes of value ("COV mode", "confirmed" or "unconfirmed") or it can read the values cyclically ("polling mode").
SX provides a PresentValue to PresentValue mapping only.

Column	Description of the Source and Destination sections
Device Instance	Instance number of the BACnet device object.
Object type	Type of BACnet object
Object Instance	Instance of the BACnet object.
Object name	Name of the BACnet object; if object instance is configured, too, the latter will be used.
Mode	Type of source-value subscription The following are valid: 1 = unconfirmed COV 2 = confirmed COV 3 = polling
Cycle Time	If the BACnet client subscribes to changes of value (Mode = 1 or 2), a repeat time in [s] must be defined for the subscription. If the source value is to be polled (Mode = 3), the cycle time must be specified in [s]. Unlimited lifetime – cycle time is 0 – is supported though not recommended. In case of SX BACnet Server as source, cyclic time 0 is only allowed for unconfirmed notifications.

Important

Supported source object types

- Analog Input
- Analog Value
- Binary Input
- Binary Value
- Multistate Input
- Multistate Value

Supported destination object types

- Analog Output
- Analog Value
- Binary Output
- Binary Value
- Multistate Output
- Multistate Value

SX BACnet Client supports Device Object as source object, too. In this case SystemStatus is used instead of PresentValue and furthermore only polling is possible and an object name for identification is not allowed, because a device object does not support the WhoHas service for itself. The read value is shifted by 2 and forwarded, the value 1 is written if source device object cannot be read. With this approach it is possible, to use a multistate object to monitor device state.

Note

A value will be written to destination object whenever a read response or a COV notification is received. The write is independent of the received value; there is no COV like behavior in SX BACnet client.
Every COV resubscription of a value is answered with a COV notification, therefore the value is written to destination more often as expected by pure COV semantic. To avoid too much load at one point in time COV lifetime is modified by a random amount it is not exactly predictable when a resubscribe will happen, but the upper bound COV lifetime will not exceeded.

Note

It is not possible to use a source object for polling and COV simultaneously. Every update of source object will be forwarded to target objects. Even if source object does not change its value, a new poll cycle or a COV resubscription will be forwarded to target objects.

5.13 BSTR Hash sheet

	A	B	C	D	E	F
1	*VT_BSTR Hashing-Table*					
2	*Defines the original BSTR-Values with their corresponding Hash-Values, which can be recognised from SX*					
3						
4	**BSTR-Value**	**Hash-Value**				
5	*String*	*Decimal-Value*				
6	"up and running"	1105260163				
7	"comm fault"	3067959514				
8	"SX OPEN"	545037869				
9	UP	3116673133				
10	DOWN	3713425771				

Figure 5-41

SX Open can deal with **VT_BSTR** items and map a finite set of different texts to [PrVal] of e.g. a multistate input object. This mapping is done by assigning every string of interest a hash value and using these hashes in a mapping function.

Calculating of hash values is performed by SX Configurator during compilation and the resulting value is stored in **BSTR Hash** sheet.

Column	**Description** of the Source and Destination sections
BSTR Value	A string to map to a hash value. If the string contains blanks it must be surrounded by quotes.
Hash Value	The resulting hash value.

5.14 Upload of properties from a running system

SX Configurator V2.1 supports the upload of properties from a running system, e.g. if a configuration shall be extended and changes of properties done by a customer must be preserved.

An upload must be accomplished in following order:

Column	Step	Procedure
Stop the SXOpen Services (**Windows 7 or later)**	**1**	If your operating system is Windows 7 or later and you have installed SXOpen as Service, you must first stop the SXOpen Services in the "Services"-Window of the control panel. When all SX services have stopped, execute "Desigo SXOpen start" window" Otherwise skip this step.
Get current values	**2**	Start **SX Upload** tool on target system.
Copy upload file to SX OPEN folder (optional)	**3**	This step is only needed if runtime system does not run on the same PC as SX Configurator.
Press Import button 	**4**	SX Configurator reads the upload file and inserts the values in **Configuration** sheet.
Configuration	**5**	Proceed with configuration.
Start the SXOpen Services (**Windows 7 or later)**	**6**	If you have done Step 1, execute "Desigo SXOpen stop window", till all SX programs have finished, then execute "Desigo SXOpen start service". Otherwise skip this step.

6 Messages

A number of messages appear in the OPC import and SX export process. Some of these are self-explanatory, and others are described below.

In the data export process, unused information is deleted automatically, to reduce the quantity of information on the screen.

Message	Description	Action
BACnet object definition is incomplete and will be ignored: clear all occurrences like this?	This information will be restored with the next OPC import.	**Yes:** = Deletes all cells containing unwanted information **No** = Add empty BACnet names or keynames only?
BACnet names and/or keynames already exist. Overwrite all?		**Yes** = All BACnet **Object Names** are reset to the OPC server information **No** = Add empty BACnet names or keynames only?
Add empty BACnet names or keynames only?	In all cases, however, other information that was removed will be restored.	**Yes** = All blanks are filled with names **No** = No names are changed

7 Examples

7.1 Mapping Limits from OPC

The following example shows the mapping of OPC items to an Analog Input Object in BACnet.

Assume the following OPC items with associated meaning:

Temperature	*actual measured value*
HighLimit	*allowed high limit of temperature*
LowLimit	*allowed low limit of temperature*

The three OPC items can be mapped to an Analog Input Object as follows.
Hereby BACnet-Properties such as highlimit, lowlimit, presentvalue are used.

Item Name	BACnet Object Type	Object Instance	Object Name	BACnet Property	Destination	High Limit	Low Limit
`Temperature`	`AI`	`1`	`TEMP`	`presentvalue`	`BACnet`	`X`	`X`
`HighLimit`	`AI`	`1`	`TEMP`	`highlimit`	`Twoway`		
`LowLimit`	`AI`	`1`	`TEMP`	`lowlimit`	`Twoway`		

To assign these three lines to one BACnet object, the same BACnet object type and object instance must be used.
SX Configurator expects the present value as the property of a new object therefore the present value mapping must be the first line

7.2 Mapping VT_BSTR Items

VT_BSTR items are mapped to multistate object most of the time. BACnet state texts are referenced by Present_Value and a mapping function is needed to map OPC item text to BACnet Present_Value and vice versa.

SX Configurator has intrinsic rules to allow an easy use of VT_BSTR items when mapped to multistate objects. Just assign a state text to multistate object and SX Configurator will fill hash table and mapping function.

If VT_BSTR items shall be mapped to analog objects hash table must be filled manually by performing following steps:

1. Define the set of possible strings which will be received via OPC item

	A	B
1	***VT_BSTR Hashing-Table***	
2	*Defines the original BSTR-Values with their corresponding Hash-Values*	
3		
4	**BSTR-Value**	**Hash-Value**
5	*String*	*Decimal-Value*
6	on	1602155427
7	level1	3082064691
8	level2	201844118
9	level3	2635075662
10	safety	2407380499
11	unknown	3300123220
12	off	504187154

Figure 7-1

The Hash-Value itself will be calculated by SX Configurator when compiling configuration and must not be calculated and entered manually.

2. Define for every text an associated BACnet state text, e.g. just use the same text.

	A	B	C	D	
1	***State texts***				
2	*Defines the texts for the Multistate objects 'stateText' property (NOTE: Do not change the 90*				
3		**State Text Information**			
4	**Ref**	1	2	3	
5	*Number*	*State Nbr Text*	*State Nbr Text*	*State Nbr Text*	
6	9013	Inactive	Active		
7	9014	Inactive	Active		
8	9019	Inactive	Active		
9	8001	Tic	Tac		
10	8002	started	normal operation	reading objects	SX Frar
11	8003	started	normal operation	not configured	reading
12	8004	started	normal operation	not configured	missing
13	8005	started	normal operation	wait for subsystem	mappin
14	8006	down	instable	instable	up
15	8007	Timeout	Operational	Operational Read Only	Downlo
16	9000	S1	S2	S3	S4
17	9001	on	level1	level2	level3
18					
19					

Figure 7-2

3. Map VT_BSTR to BACnet Present_Value using defined strings and mapping function *BstrHash2MultiState*

BstrHash2M	7	H<on>	H<level1>	H<level2>	H<level3>	H<safety>	H<unknov	H<off>

- H indicates to use the Hash-Value of provided string

7.3 Defining a Calendar Object

Calendar objects are created as container only. There will be no calendar entries inserted by SX-Configurator. Do add a calendar object to SX-Configurator a new line must be inserted in Configuration Sheet.

Item Name	OPC Access Rights	OPC Datatype	VT_ARRAY index	OPC Server Ref	OPC Group Update Rate	Item Set Ref	Device Set Ref
			I..	Ref.	ms	Ref.	Ref.
SX-OPEN.internal1	RW	VT_UI4		1	2000		

Figure 7-3

Calendar objects are SX Open internal objects, there is no OPC item associated with a calendar object. Nevertheless SX-Configurator needs an OPC item for every BACnet object defined in Configuration Sheet. To solve this situation SX-Configurator provides internal items which can be used for this purpose and which will be not copied into SXOPCclient.csv.

SX-OPEN.internal1	Name of internal item
OPC Server Ref	Server Id of Supervisor (which is based on internal items, too)

BACnet				
BACnet Device Ref	BACnet Object Type	Object Instance	Keyname	BACnet Object Name
Ref.	Nickname or Name	I..	String	String
611	CAL	62	calendar	calendar

Figure 7-4

Assigning calendar object is straightforward. Just use object type CAL and assign BACnet Server, object instance, object name and keyname.

Function info				
Avoid Default Mapping	Use OPC Quality	Function Name	Parameter 1	Parameter 2
I or o	I or on	String from Functions sheet		
		SXVirtual	0	

Figure 7-5

As a last step entries in SXmapping.csv must be suppressed. This is done by using the special mapping function *SXVirtual* which is a trigger not to emit mapping entries for this row in Configuration sheet. Parameter number 1 to 9 are reserved for Supervisor objects, every othere parameter value is feasible.

After compiling this configuration and starting SX_Open a calendar object will be available and can be filled with value from Desigo Insight.

7.4 Assigning a Schedule to an Object

- In the Configuration table, select a BACnet object (a line).
- Assign a schedule to it by entering an index number in the Configuration table / row "Schedule Object Instance".
- After the translation a new line appears in the Schedule sheet, which can be further adjusted. The index number can be found in the Configuration table / row "Schedule Object Instance", it must be unique, and it must be incremented for each schedule instance. This index number is the link between the schedule and the BACnet object.

8 FAQ

What is the meaning of SX object member eventEnable?

SX object member *eventEnable* is the counterpart of BACnet property *event-enable* and is entered as a number that will be decoded as follows:

- Bit 0: TO_OFFNORMAL
- Bit 1: TO_FAULT
- Bit 2: TO_NORMAL

The coding is hidden in SX Configurator VBA code. *EventEnable* must be inserted in the *BACnet properties* sheet. *EventEnable* will be decomposed into *ToOffNormalEventEnable*, *ToNormalEventEnable* and *ToFaultEventEnable* by SX Configurator and this decomposition will be inserted into *SXBacNet.csv*.

How do I configure a binary-output object without an alarm-value?

Binary-output objects don't possess an alarm-value property. So this column must be left blank in SX Configurator.

What's the meaning of "Alarm defined but no alarm message text"?

SX Configurator complains about a missing text in the *message text* column. This text is an additional parameter for an event-notification telegram and may give a detailed description of the event.

What's the meaning of "Output object doesn't have alarm/fault values; use EventEnable property to enable alarms"?

Binary-output objects don't possess an alarm-value property. Configuring such a property leads to the above error message, because binary-output objects don't have any events enabled by default. The same is true for multistate-output objects. Additionally such objects don't possess a fault-values property, too.

How do I change BACnet object type and BACnet device when reading OPC items?

SX Configurator assigns BACnet object types based on configuration data located in **BACnet Device sheet** and hidden **OPCImportTypeMapping sheet**.

BACnet device

All imported items will be assigned to device **AUTOIMPORT**. If AUTOIMPORT is not found it will be inserted.

OPCImportTypeMapping

This sheet uses three columns to map an OPC item type to a BACnet object type

- OPC item type
- OPC access rights
- BACnet object type to select

For example to map VT_UI2 items to analog value independent of access rights the following rules must be inserted:

	A	B	C	D
1	VT_UI2	R	AVALOP	
2	VT_UI2	RW	AVALOP	
3	VT_UI2	W	AVALOP	
4				

Index

Siemens Switzerland Ltd
Building Technologies Division
International Headquarters
Gubelstrasse 22
6301 Zug
Switzerland
Tel. +41 41 724 24 24
www.siemens.com/buildingtechnologies