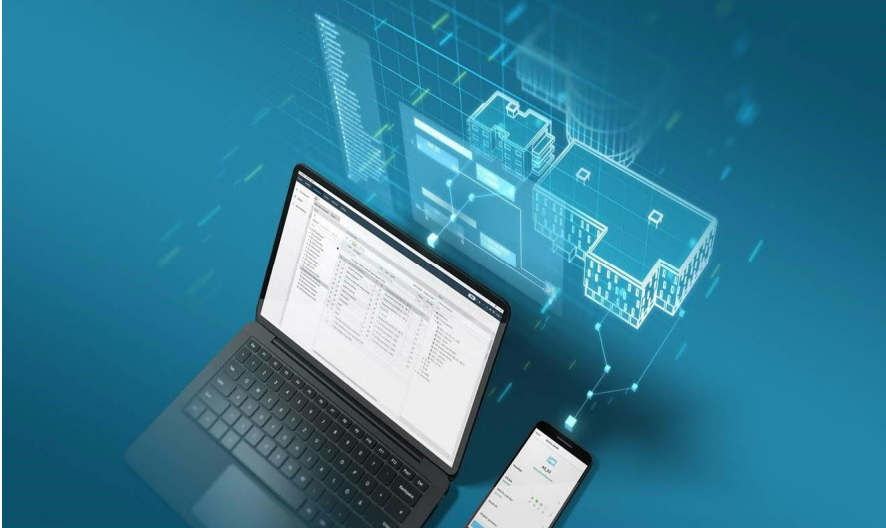


SIEMENS



Desigo™

Building automation system 6.3

Technical Principles

Copyright

Delivery and technical specifications subject to change.

Any form of reproduction, dissemination, copying, disclosure, modification, distribution and/or publication of this document is strictly prohibited unless permitted expressly. Infringements will lead to compensation. All rights, including rights created by patent grant or registration of a utility model or design are reserved.

Published by:

Siemens Switzerland Ltd.

Smart Infrastructure

Global Headquarters

Theilerstrasse 1a

CH-6300 Zug

Tel. +41 58 724-2424

www.siemens.com/buildingtechnologies

Edition: 2021-06-01

Document ID: CM110664en_08

© Siemens Switzerland Ltd, 2015

Table of Contents

| | | |
|----------|---|------------|
| 1 | Cyber security disclaimer | 9 |
| 2 | Preconditions of this document | 10 |
| 3 | System overview | 11 |
| 3.1 | Management level..... | 12 |
| 3.2 | Automation level | 14 |
| 3.3 | Room automation..... | 16 |
| 3.4 | Field level | 17 |
| 3.5 | Desigo Open..... | 18 |
| 3.6 | Workflow and tools..... | 19 |
| 3.7 | Topologies | 20 |
| 3.8 | Communication principles | 22 |
| 3.9 | Data maintenance..... | 24 |
| 3.10 | Views..... | 30 |
| 4 | Desigo workflow, tools and programming | 33 |
| 4.1 | Coverage of the technical process..... | 33 |
| 4.2 | Coverage of the system | 35 |
| 4.3 | Main tasks | 37 |
| 4.4 | Tools for different roles..... | 40 |
| 4.5 | Working with libraries | 41 |
| 4.6 | Working in parallel and subcontracting | 42 |
| 4.7 | Workflow for primary systems..... | 43 |
| 4.8 | Workflow for room automation classic | 44 |
| 4.9 | Workflow for Desigo room automation..... | 44 |
| 4.10 | Desigo Configuration Module (DCM) | 45 |
| 4.11 | Desigo Xworks Plus (XWP)..... | 46 |
| 4.12 | Desigo Automation Building Tool (ABT)..... | 51 |
| 4.13 | Programming in D-MAP | 53 |
| 5 | Control concept | 56 |
| 5.1 | Control concept and control blocks..... | 61 |
| 5.2 | Local control design | 69 |
| 5.3 | Superposed plant controls..... | 72 |
| 5.4 | Closed-loop control strategy..... | 87 |
| 5.5 | Desigo room automation | 95 |
| 6 | Technical view | 111 |
| 6.1 | Standard plant structures | 111 |
| 6.2 | Technical text labels..... | 114 |
| 7 | Global objects and functions | 118 |
| 7.1 | Ensuring data consistency..... | 118 |
| 7.2 | Roles in the system..... | 119 |
| 7.3 | Life check | 120 |
| 7.4 | Time synchronization | 121 |
| 7.5 | Examples of global objects..... | 121 |

| | | |
|-----------|--|------------|
| 8 | Events and COV reporting | 125 |
| 8.1 | Sources and causes of system events | 125 |
| 8.2 | Routing system events | 125 |
| 8.3 | Sources and causes of COVs | 126 |
| 8.4 | COV reporting | 126 |
| 9 | Alarm management | 129 |
| 9.1 | Alarm sources | 129 |
| 9.2 | Alarm example | 131 |
| 9.3 | Effects of BACnet properties on alarm response | 133 |
| 9.4 | Alarm response of the function blocks | 140 |
| 9.5 | Alarm functions | 147 |
| 9.6 | Alarm management by notification class | 149 |
| 9.7 | Alarm routing over the network | 153 |
| 9.8 | Alarm queuing | 155 |
| 9.9 | Common alarms | 157 |
| 9.10 | Alarm suppression | 158 |
| 9.11 | Alarm message texts | 159 |
| 10 | Calendars and schedulers | 161 |
| 10.1 | Schedule | 162 |
| 10.2 | Calendar | 166 |
| 10.3 | Wildcards | 167 |
| 10.4 | Alarm messages | 167 |
| 11 | Trending | 168 |
| 11.1 | Trend functions | 168 |
| 11.2 | Editing parameters | 170 |
| 11.3 | Processing trend data in Desigo CC | 171 |
| 12 | Reports | 172 |
| 13 | Data storage | 173 |
| 13.1 | Data categories | 173 |
| 13.2 | Program data | 173 |
| 13.3 | Libraries | 174 |
| 13.4 | Project data | 175 |
| 13.5 | Plant data | 176 |
| 13.6 | Data transfer processes | 176 |
| 13.7 | Texts | 178 |
| 14 | Network architecture | 179 |
| 14.1 | BACnet architecture (MLN & ALN) | 179 |
| 14.2 | LonWorks architecture (ALN) | 191 |
| 14.3 | KNX architecture (ALN) | 192 |
| 14.4 | KNX PL-Link architecture (FLN) | 193 |
| 14.5 | DALI architecture (FLN) | 194 |
| 15 | Remote access | 196 |
| 15.1 | Remote access methods | 196 |
| 15.2 | Choosing a suitable access technology | 197 |
| 15.3 | Technical details | 198 |
| 16 | Management platform | 200 |

| | | |
|-----------|--|------------|
| 16.1 | User functions..... | 202 |
| 16.2 | Main components..... | 204 |
| 16.3 | Access and security | 205 |
| 16.4 | Event management..... | 206 |
| 16.5 | Installation, setup and engineering | 207 |
| 16.6 | Graphics libraries..... | 209 |
| 16.7 | Graphics engineering | 210 |
| 16.8 | Virtual environment | 212 |
| 17 | Desigo Control Point..... | 213 |
| 17.1 | Functions | 214 |
| 18 | Automation stations | 218 |
| 18.1 | Device object | 219 |
| 18.2 | Device info object..... | 220 |
| 18.3 | Error sources and monitoring functions | 221 |
| 18.4 | Operating states..... | 222 |
| 18.5 | Data storage | 226 |
| 19 | Logical I/O blocks | 228 |
| 19.1 | General functions..... | 228 |
| 19.2 | Input blocks..... | 244 |
| 19.3 | Output blocks..... | 246 |
| 19.4 | Value objects | 249 |
| 19.5 | Value objects for operation..... | 252 |
| 19.6 | Addressing the I/O blocks | 252 |
| 19.7 | Discipline I/Os..... | 261 |
| 19.8 | Reliability table..... | 262 |
| 19.9 | Slope [Slpe] and Intercept [Icpt]..... | 263 |
| 19.10 | Addressing entries for PXC...-U, PTM and P-Bus..... | 269 |
| 20 | Room automation..... | 275 |
| 20.1 | Desigo room automation | 275 |
| 20.1.1 | Configurable..... | 276 |
| 20.1.2 | Programmable | 285 |
| 20.1.3 | Rooms and room segments..... | 288 |
| 20.1.4 | Central control functions and grouping..... | 289 |
| 20.1.5 | Desigo room automation and the management level..... | 290 |
| 20.1.6 | Desigo room automation and the automation level..... | 290 |
| 20.2 | Desigo RXB | 290 |
| 20.2.1 | Product range overview..... | 291 |
| 20.2.2 | RXB and the management level | 292 |
| 20.2.3 | RXB and the automation level | 293 |
| 20.2.4 | RXB applications..... | 293 |
| 20.2.5 | Mapping RXB in the PX KNX system controller | 293 |
| 21 | Desigo Open..... | 294 |
| 21.1 | Integration on management level | 295 |
| 21.2 | Integration on automation level..... | 296 |
| 21.3 | Integration on field level | 298 |
| 21.4 | Integration on room level..... | 299 |

| | | |
|-----------|--|------------|
| 22 | System configuration | 300 |
| 22.1 | Technical limits and limit values | 302 |
| 22.2 | Maximum number of elements in a network area..... | 303 |
| 22.3 | Desigo room automation system function group limits | 304 |
| 22.4 | Devices..... | 305 |
| 22.4.1 | PXC..D automation stations / system controllers..... | 305 |
| 22.4.2 | LonWorks system controllers..... | 308 |
| 22.4.3 | Automation stations with LonWorks integration..... | 308 |
| 22.4.4 | PX Open integration (PXC001.D/-E.D) | 309 |
| 22.4.5 | PX Open integration (PXC001.D/-E.D + PXA40-RS1)..... | 309 |
| 22.4.6 | PX Open integration (PXC001.D/-E.D + PXA40-RS2)..... | 309 |
| 22.4.7 | PX KNX integration (PXC001.D/-E.D)..... | 309 |
| 22.4.8 | TX Open integration (TXI1/2/2-S.OPEN) | 310 |
| 22.4.9 | Number of data points on Desigo room automation stations | 310 |
| 22.4.10 | Number of data points for PXC3..... | 312 |
| 22.4.11 | Number of data points for DXR1..... | 312 |
| 22.4.12 | Number of data points for DXR2..... | 313 |
| 22.4.13 | PXM20 operator unit | 314 |
| 22.4.14 | PXM10 operator unit | 314 |
| 22.4.15 | Desigo Control Point | 314 |
| 22.4.16 | PXG3.L and PXG3.M BACnet routers | 317 |
| 22.4.17 | SX OPC | 318 |
| 22.4.18 | Desigo CC | 318 |
| 22.4.19 | Desigo Insight | 318 |
| 22.4.20 | Desigo Xworks Plus (XWP) | 318 |
| 22.4.21 | Desigo Automation Building Tool (ABT)..... | 319 |
| 22.5 | Applications | 319 |
| 22.5.1 | Peak Demand Limiting (PDL) | 319 |
| 23 | Compatibility..... | 320 |
| 23.1 | Desigo version compatibility definition | 320 |
| 23.2 | Desigo system compatibility basics | 321 |
| 23.2.1 | Compatibility with BACnet standard..... | 321 |
| 23.2.2 | Compatibility with operating systems..... | 322 |
| 23.2.3 | Compatibility with SQL servers..... | 323 |
| 23.2.4 | Compatibility with Microsoft Office..... | 323 |
| 23.2.5 | Compatibility with web browsers..... | 323 |
| 23.2.6 | Compatibility with ABT Go..... | 324 |
| 23.2.7 | Compatibility with VMware (virtual infrastructure)..... | 324 |
| 23.2.8 | Compatibility of software/libraries on the same PC | 324 |
| 23.2.9 | Hardware and firmware compatibility | 324 |
| 23.2.10 | Backward compatibility | 325 |
| 23.2.11 | Engineering compatibility..... | 325 |
| 23.2.12 | Compatibility with Desigo Configuration Module (DCM) | 325 |
| 23.2.13 | Compatibility with Desigo PX / Desigo room automation | 325 |
| 23.2.14 | Compatibility with Desigo RX tool..... | 325 |
| 23.2.15 | Compatibility with TX-I/O | 326 |
| 23.2.16 | Compatibility with TX Open | 326 |

| | | |
|-----------|---|------------|
| 23.2.17 | KNX PL-Link devices in ABT Site and ABT Pro | 327 |
| 23.3 | Desigo Control Point | 338 |
| 23.3.1 | Compatibility with earlier systems..... | 338 |
| 23.3.2 | Compatibility with earlier devices..... | 338 |
| 23.3.3 | Supported browsers | 339 |
| 23.4 | Upgrading to Desigo V6.3 | 340 |
| 23.4.1 | Upgrading for Tool and Localization Managers in the Regional Companies (RC) | 340 |
| 23.4.2 | Upgrading for Project Engineers..... | 340 |
| 23.4.3 | Upgrading restrictions | 342 |
| 23.5 | Siemens WEoF clients | 342 |
| 23.6 | Migration compatibility..... | 343 |
| 23.7 | Hardware requirements of Desigo software products..... | 344 |
| 24 | Desigo PXC4 and PXC5 | 346 |
| 25 | Compatibility of Desigo V6.3 with PXC4 and PXC5..... | 347 |

1 Cyber security disclaimer

Siemens provides a portfolio of products, solutions, systems and services that includes security functions that support the secure operation of plants, systems, machines and networks. In the field of Building Technologies, this includes building automation and control, fire safety, security management as well as physical security systems.

In order to protect plants, systems, machines and networks against cyber threats, it is necessary to implement – and continuously maintain – a holistic, state-of-the-art security concept. Siemens' portfolio only forms one element of such a concept.

You are responsible for preventing unauthorized access to your plants, systems, machines and networks which should only be connected to an enterprise network or the internet if and to the extent such a connection is necessary and only when appropriate security measures (e.g. firewalls and/or network segmentation) are in place. Additionally, Siemens' guidance on appropriate security measures should be taken into account. For additional information, please contact your Siemens sales representative or visit <https://www.siemens.com/global/en/home/company/topic-areas/future-of-manufacturing/industrial-security.html>.

Siemens' portfolio undergoes continuous development to make it more secure. Siemens strongly recommends that updates are applied as soon as they are available and that the latest versions are used. Use of versions that are no longer supported, and failure to apply the latest updates may increase your exposure to cyber threats. Siemens strongly recommends to comply with security advisories on the latest security threats, patches and other related measures, published, among others, under <https://www.siemens.com/cert/en/cert-security-advisories.htm>.

2 Preconditions of this document

IT security

Building automation and control systems such as Desigo are increasingly integrated into a building's IT infrastructure and will often also be remotely accessible. Besides using the IT security features of the various products, it's very important to implement an IT secure integration into the site's IT infrastructure. For guidelines for such an IT secure integration, see *IT Security in Desigo Installations* (CM110663). These guidelines are binding and must be implemented in every Desigo project.

Furthermore, the usual rules and best practice procedures from the IT world must also be observed to achieve a high protection level for the building automation and control system and the customer's IT infrastructure.

Compatibility with third party products

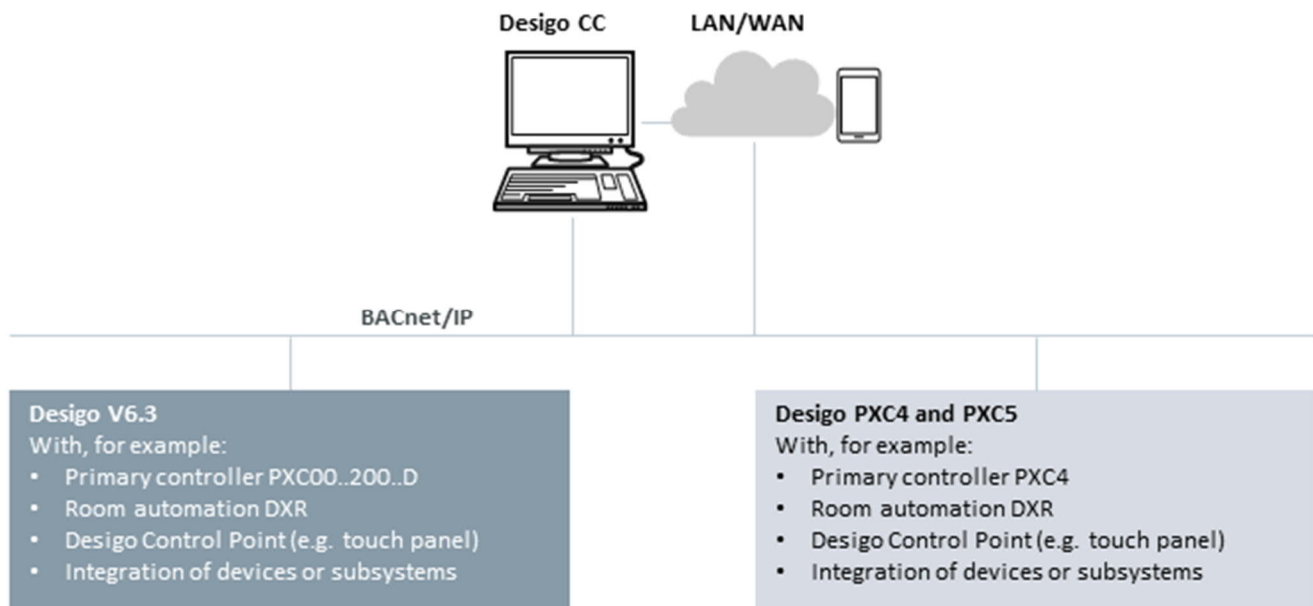
This document lists various versions of third party hardware and software, Siemens has tested the listed versions for compatibility with this version of the Desigo release and was satisfied with the results.

Nevertheless, Siemens assumes no warranty or liability for any compatibility, be it backward, current and under no circumstances for future versions thereof as such compatibility is extremely dependent on the interaction of the system as a whole and the different components interacting. If assurance on compatibility is needed, then please inquire about a specific maintenance agreement for your system or solution.

Document structure

This document is split into two parts:

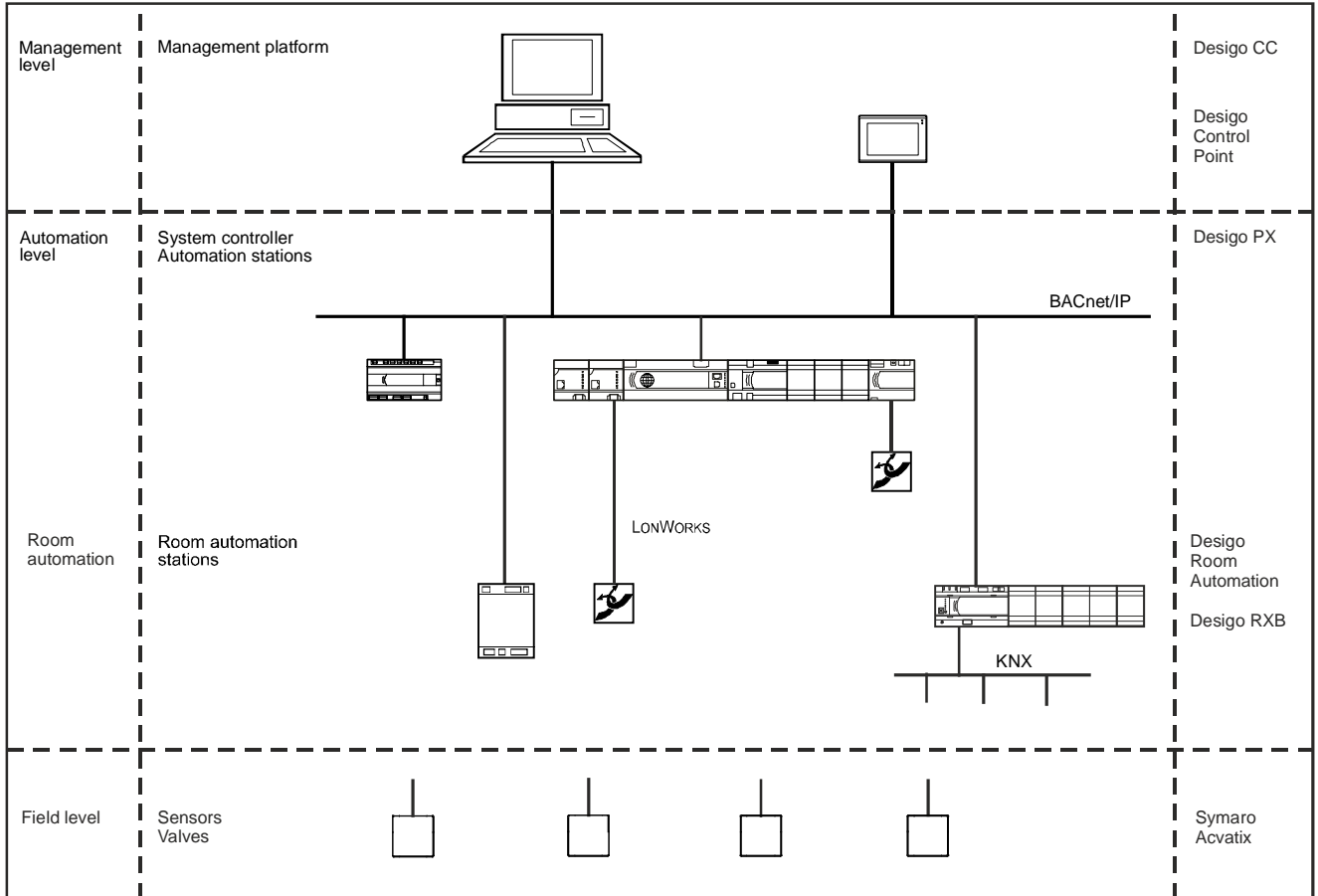
- Chapter 3 to 23: Desigo System V6.3
- Chapter 24: Desigo PXC4 and PXC5 range overview and planning overview
- Chapter 25: Compatibility between Desigo System V6.3 and Desigo PXC4 and PXC5



3 System overview

The Designo building automation and control system has three levels:

- Management level
- Automation level
- Field level



3.1 Management level

Operation and monitoring

The key functions at the management level are operation and monitoring of the plant, including:

- Graphics-based operation of the plant
- Cross-site alarm generation and alarm transfer
- Maintenance of a long-term log
- Storage and graphical display of trend data
- Graphics-based operation of time schedules
- Display, navigation and modification of data objects, which are displayed in a hierarchical tree structure
- Visual monitoring of the operation of primary plants (monitoring to reduce energy consumption and wear and tear)
- Visual monitoring of the rooms (HVAC, lights and blinds)
- Reporting function including energy reports
- Centralized time control and calendar functions
- Event program: Triggering system reactions based on system events

What is operation and monitoring?

Operation and monitoring encompasses all the interaction between a user and the plant via the building automation and control system.

| Task | Activity |
|--|--|
| Observing the operational status of the plant or building | Reading current values of all process variables, data objects and parameter settings Receiving and acknowledging alarms Overview of all pending alarms Recording and analyzing trends |
| Observing the operational status of the building automation and control system | Overview of failed automation stations and network interruptions Signaling of anormal hardware or software states in an automation station or in the associated field devices |
| Manipulating the operational status of the plant or building | Modifying parameter settings (e.g., setpoints of control programs) Setting values for physical outputs of automation stations Modifying system and management objects, especially calendars and time schedules |

Devices for operation and monitoring

The following devices let you operate and monitor the system:

- Desigo CC management platform either locally and/or with web operation
- PXM touch panels and operating devices

Operation and monitoring types

There are four operation and monitoring types:

- Generic operation
- Limited (station-specific) generic operation
You can limit the generic view to one or more selected automation stations (including alarm display).
- Engineered (project-specific) operation
You can generate a project-specific view in the engineering phase.
- Limited (user-specific) operation in Desigo CC

Management platform

Desigo CC can be installed on one computer, with full server and client functionality, or on several separate computers. Web Clients, Windows App (ClickOnce) Clients and installed Clients can be added.

Remote management

Desigo CC can operate and monitor the automation level via a public network.

3.2 Automation level

The Desigo PX automation system meets all the requirements for the control and monitoring of heating, ventilation, air conditioning systems and other building services. Desigo PX with its programmable automation stations and graded range of operator units is a scalable and open system.

D-MAP programming language

The starting point for the engineering of the application functions is the range of user-friendly application blocks and function blocks in the D-MAP (Desigo Modular Application Programming) programming language. D-MAP is optimized for applications for technical building installations and is based on the IEC 1131 standard. The graphical user interface of the Xworks Plus (XWP) engineering software ensures an efficient approach.

System functions

All PX automation stations have comprehensive system functions, such as alarm management, time schedules, trend histories, time synchronisation, global data distribution, and life check, that work completely autonomously.

BACnet communication for maximum openness

Devices on the automation level communicate with each other and with the management platform and the operating devices via the BACnet protocol.

The use of BACnet/IP or BACnet/LonTalk underlines the openness of the system and allows the easy integration of systems and components from third-party manufacturers.

Automation stations and system controllers

PX Modular

The Desigo PX range of programmable modular automation stations provides maximum flexibility for controlling and monitoring building services. Comprehensive system functions, such as alarm management, time schedules and trend histories, meet all requirements for technical building installations.

The PXX-Lxx extension modules let you connect LonWorks devices and third-party devices.

The PXX-PBUS extension module lets you integrate PTM IO modules.

The PXA40-Tx option modules provide functions, such as web operation.

TX-I/O

Desigo TX-I/O modules provide the interface between PX Modular and the field level devices, the actuators and sensors.

A range of configurable and flexible I/O modules are available for signalling, measuring, metering, switching and controlling.

Some modules can be manually operated according to ISO 16484, and have an LCD display with configurable LEDs.

The integrated isolating-terminals facilitate the hardware test during commissioning.

TX Open

TXix.OPEN lets you integrate third-party systems, such as M-Bus meters, pumps (Grundfos, Wilo) and variable speed drives (Siemens G120P), and connect intelligent aggregates, e.g., chillers, via the Modbus protocol.

PX Compact

The Desigo PX range of programmable compact automation stations with integrated I/Os provides optimized solutions for small to mid-sized technical building installations. Comprehensive system functions, such as alarm management, time schedules and trend histories, meet all requirements for technical building installations.

PX Compact with island bus

The easy to install PXC22.1...D and PXC36.1...D compact automation stations offer increased flexibility due to the island bus.

Properties:

- Up to 52 physical I/Os
- Integration of up to 5 subsystems, such as Modbus, M-bus, with up to 400 data points

PX Open

PX Open system controllers let you integrate third-party devices via Modbus, M-Bus, KNX and other protocols. System functions, such as alarm management, time schedules, trend data storage and flexible programming are available.

Web interfaces and touch panels

The various operator units cover all the various requirements in terms of location and function.

PXG3.W100-1 and PXG3.W200-1 web interface

The BACnet/ IP Web interface permits local and remote operation of Desigo primary and room automation stations as well as third-party BACnet/IP devices. The products PXG3.W100-1 and PXG3.W200-1 vary in functionality as well as permissible system limits.

Use:

- Remote access to the plant
- Local operation using third-party devices

PXM30-1, PXM40-1, and PXM50-1 touch panels

The TCP/IP touch panels are used on projects that require multiple touch panels or third-party devices to operate the same system data. The central web interface PXG3.Wx00-1 can connect multiple devices.

Use:

- Multiple touch panels on a plant room, conference room, open office spaces, etc.
- A touch panel in the plant room for on-site operation, remote access via third-party devices

PXM30-E, PXM40-E, and PXM50-E touch panels

The BACnet/IP touch panels can be connected directly to the BACnet network. No additional devices required to operate the Desigo system.

In addition, the BACnet/IP touch panel also includes a web interface for remote operation using a standard web browser on third-party devices. The functionality of the web interface essentially matches the functionality of the PXG3.W100-1, differing only in the permissible system limits. The integrated web access increases the competitiveness of Desigo on smaller projects since there are no additional costs and installation effort.

Use:

- Very small projects: One touch panel for on-site operation of a primary plant. Remote access occurs directly via the integrated web interface on the touch panel.
- Large project with multiple, decentralized touch panels. The central, cross-building operation takes place on Desigo CC.

3.3 Room automation

The room automation is part of the automation level. The room automation includes devices for the control functions within a room.

There are RX room controllers and Desigo PXC3/DXR2 room automation stations.

The PXC3/DXR2 room automation stations have the following functions:

- Measuring, controlling and processing of I/O signals
- Logging trend data
- Monitoring process variables and generating alarms
- Acknowledging and resetting alarms
- Monitoring process variables for value changes
- Exchanging data with clients and other automation stations
- Monitoring hardware and software functions and generating events in case of faults or errors
- Processing BACnet access for operation and monitoring of one or multiple clients
- Handling errors, e.g., during data point exchange

The PX automation stations carry out coordination functions (Desigo room automation system functions), such as time synchronisation, life check, scheduling, etc., for the room automation stations.

Desigo supports the following communication technologies:

- BACnet
- KNX technology
- DALI (Digital Addressable Lighting Interface)
- LonWorks technology (only for RX)

Desigo room automation (PXC3..)

In Desigo room automation freely programmable PXC3 room automation stations control the room climate. The Desigo room automation product range integrates several disciplines (HVAC, lighting, shading). A room automation station can cover several rooms. The room automation stations are integrated seamlessly into Desigo PX and the management level via BACnet/IP.

Buttons, sensors, and actuators are connected to the PXC3 room automation via TX-I/O modules or KNX PL-Link modules.

The KNX interface of the PXC3 room automation stations allows the direct integration of devices with KNX PL-Link and KNX S-Mode in Desigo room automation. KNX PL-Link is fully compliant with the KNX standard. The PXC3 room automation stations support plug and play functionality with automation device detection. Devices with KNX PL-Link are parameterized with the Desigo tools. The KNX commissioning software (ETS) is not needed.

The PXC3.. room automation stations have an integrated web server for IP communication with touch room operator units. Engineering access is available via the web interface.

A subset of the available TX-I/O modules can be used with the PXC3 automation station.

The DALI (Digital Addressable Lighting Interface) bus of the PXC3...A room automation station lets you integrate lighting.

The PXC3.E16A room automation station is tailored for lighting applications. It has an on-board DALI interface for integrating up to 64 ECGs (electronic control gear).

Desigo room automation (DXR2..)

The DXR2 room automation stations let you automate heating, ventilation, air conditioning, shading, and lighting for rooms.

The room automation stations communicate with each other and other system components via BACnet/IP (DXR2.E...) or BACnet MS/TP (DXR2.M...).

The room automation stations support different I/O mixes, protocols (KNX S-Mode and KNX PL-Link for IP and KNX PL-Link for MS/TP) and power supplies (240/24V). Operating devices, buttons, sensors, and actuators for lighting and shading can be connected to the room automation stations via KNX PL-Link.

The room automation stations contain preloaded applications, but are also freely programmable. A comprehensive library of proven, standardized applications is available.

The DXR2.. room automation stations have an integrated web server for IP communication with touch room operator units. Engineering access is available via the web interface.

Desigo RXB

The RXB room controllers control the room climate in individual rooms and important parameters of the applications can be configured.

The RXB room controllers communicate via KNX.

The PX KNX system controller connects the room automation devices to Desigo PX and the management level and assumes coordination functions for room automation (grouping, scheduling, demand signal exchange, peer-to-peer, etc.).

3.4 Field level

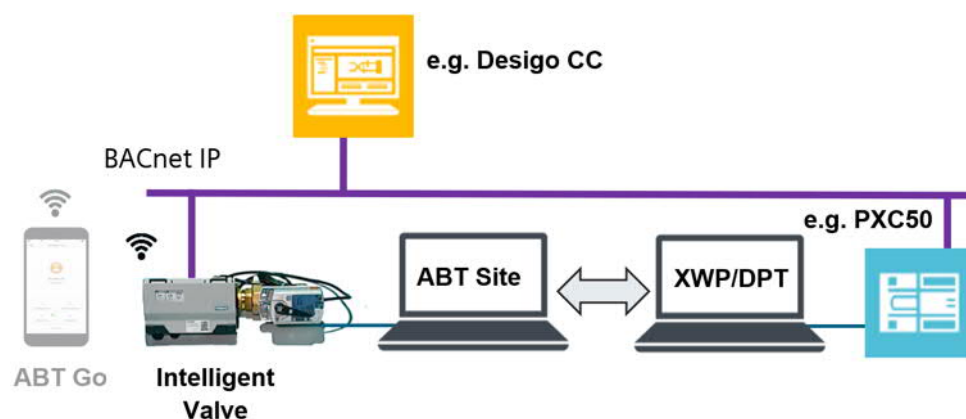
Acvatix Intelligent Valve

Acvatix Intelligent Valve is a pre-loaded device with a specific application. The Desigo tool environment supports the Intelligent Valve to allow smooth and lean BACnet/IP integration into Desigo projects.

See *Intelligent Valve Datasheet* (A6V11444716)

Use case

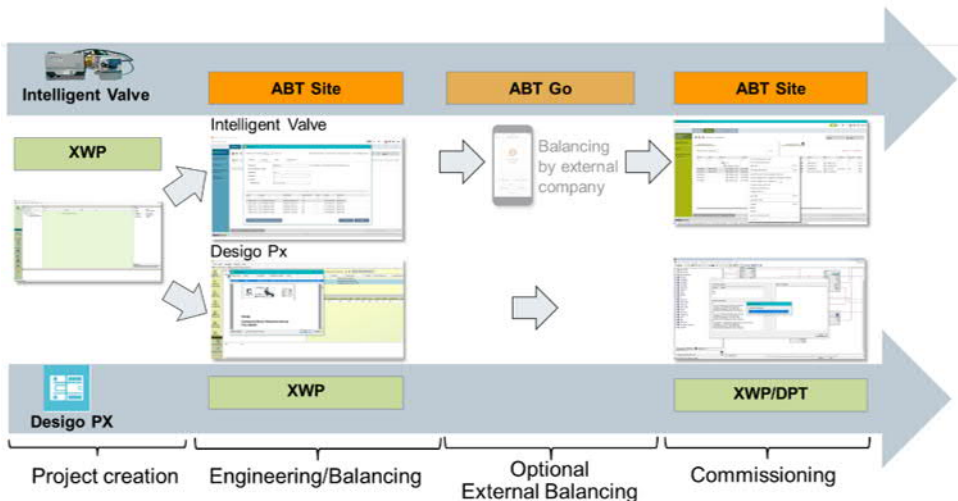
Acvatix Intelligent Valve acts as a control valve within a PX primary system. The Intelligent Valve communicates via BACnet IP.



Engineering and commissioning

Desigo Xworks Plus (XWP) in collaboration with ABT Site support engineering and commissioning of Acvatix Intelligent Valves with the following functionality:

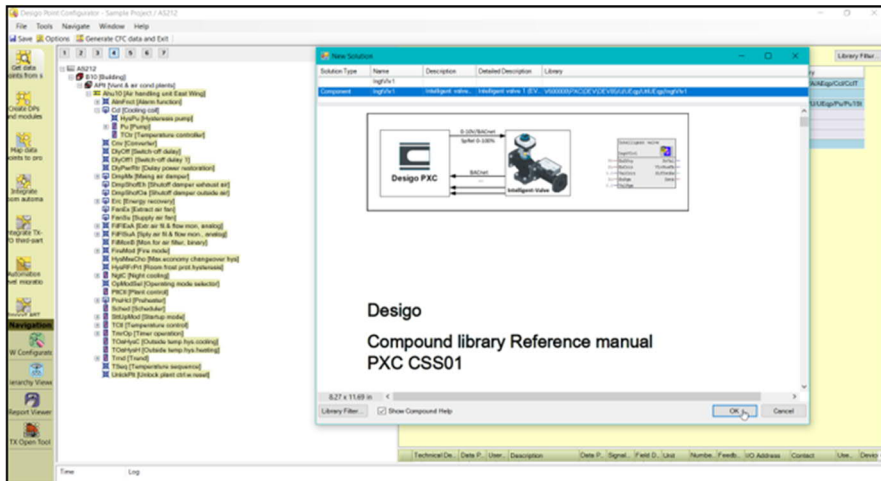
- Collaboration with Desigo Xworks Plus (XWP) for integration into Desigo PX Primary controller
- Mass instantiation of Intelligent Valve devices
- Parameterization of control and balancing functions with dedicated application type HvacFnct34 and prepared application templates
- Pairing of virtual and physical devices
- Mass download of Intelligent Valve application configurations
- Upload of run-time instance data



See *Engineering Instructions Document (A6V11572317)*.

Compound

The most sophisticated solution for new projects supports the complete workflow including the new Compound {IngtVlv1} for the Designo PX Primary controllers. This compound offers a pre-engineered solution representing the most important datapoints, a trending of selected datapoints and an evaluation of the Intelligent Valve error indication used for alarming.



See *Compound Library Reference Manual (A6V11625777)*.

3.5 Design Open

Designo Open lets you integrate devices and systems from different manufacturers into the Designo system.

Designo Open supports various protocols, e.g., OPC, Modbus, KNX/EIB, LonWorks, M-Bus, KNX, DALI, etc. for integrating energy monitoring, fire security, access control and security, power distribution, refrigeration machines, pumps, meters, variable speed drives, lighting and blinds, etc.

Regional companies can use Software Development Kits (SDKs) to develop their own solutions.

Integration on the management level

Designo CC uses BACnet, Modbus, OPC, SNMP, and RESTful web services to exchange data with third-party systems.

Integration on the automation level

PX Open system controllers let you integrate third-party devices on Modbus, M-Bus, KNX and other protocols, by converting all data into standard BACnet objects.

Integration on the field level

TXIx.OPEN lets you integrate third-party systems, such as M-Bus meters, pumps (Grundfos, Wilo) and variable speed drives (Siemens G120P), and connect intelligent aggregates, e.g., chillers, via the Modbus protocol.

3.6 Workflow and tools

The Desigo tools cover parts of the technical process and parts of the Desigo system:

- Desigo Configuration Module (DCM) lets you plan the system and determine the quantity during the sales phase.
- Xworks Plus (XWP) lets you engineer, commission and service Desigo PX system components.
- ABT Pro and ABT Site (Automation Building Tool) let you engineer, commission and service Desigo (BACnet) system components.
- PX KNX-Tool lets you commission and service PX KNX.
- Desigo CC Graphics Generator lets you automatically generate Desigo CC plant graphics using information from the System Definition Unit (SDU) and XWP.
- System Definition Unit (SDU) lets you define application texts in different languages.
- PX Open MONITOR lets you debug PX Open programs.
- TX Open tool lets you configure and commission TX Open modules.
- BIM tool lets you:
 - Commission TX-I/O modules and the Bus Interface Modules (BIM)
 - Simulate programs without I/O modules on the test rack
 - Configure the colors of the I/O status LED on the TX-I/O modules
- Desigo Automation Level Migration Tool lets you copy engineering parameters, such as I/O addresses, texts, data point parameters, PID controller parameters and trend objects, of a Visonik controller to a PX automation station.
- Desigo Point Test (DPT) lets you test data points for field devices and PX automation stations during commissioning.

Preloaded applications

Some automation stations contain preloaded applications, but are also freely programmable. A comprehensive library of proven, standardized applications is available and can be used instead of the preloaded applications.

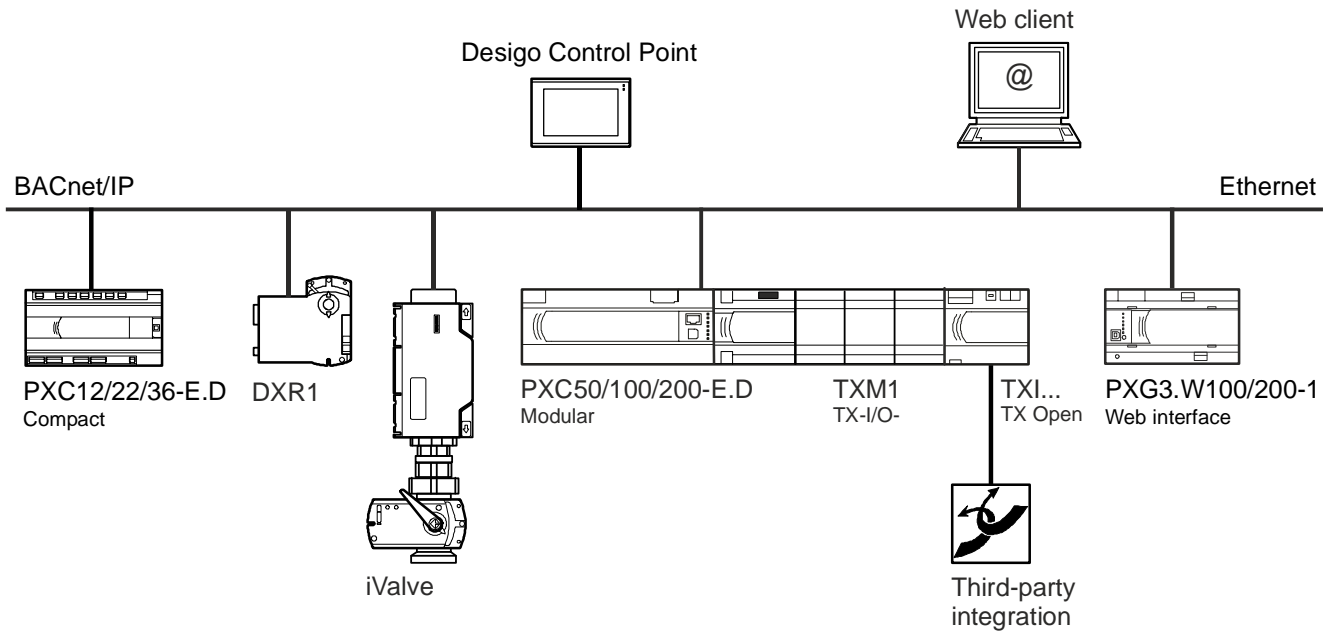
XWP to PXC communication

XWP communicates with the PX automation station via BACnet/IP or BACnet/LonTalk. The CFC or Parameter Editor can communicate online with the PX automation stations. This is a useful aid both for commissioning and testing the automation stations, and for operation and monitoring. The pin values and some attributes of the compounds and blocks can be modified online.

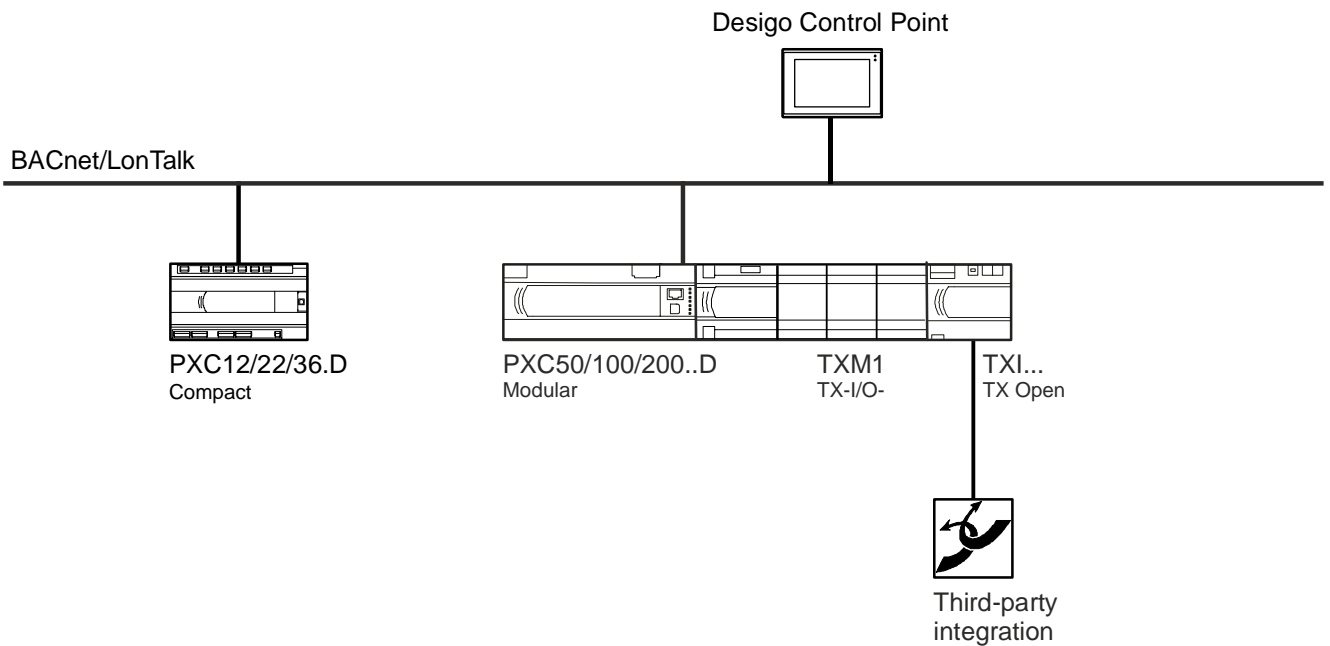
To commission a Lon-based PX automation station, XWP must be connected to the same LonWorks network as the automation station. The program or program changes can be downloaded via BACnet router or PTP connection, which can also be used for monitoring and operation. The functionality to configure and commission the BACnet router is integrated in the XWP Network Configurator.

3.7 Topologies

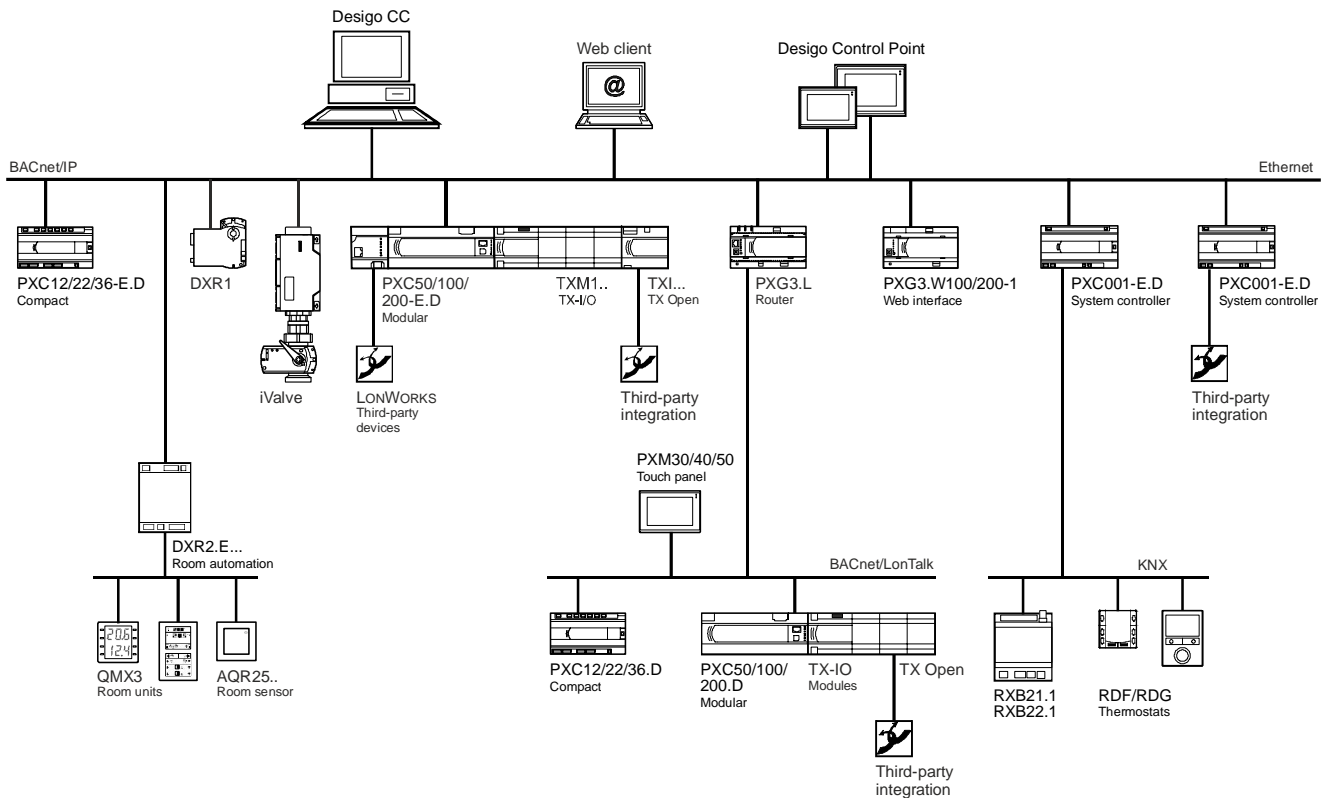
Small system on BACnet/IP



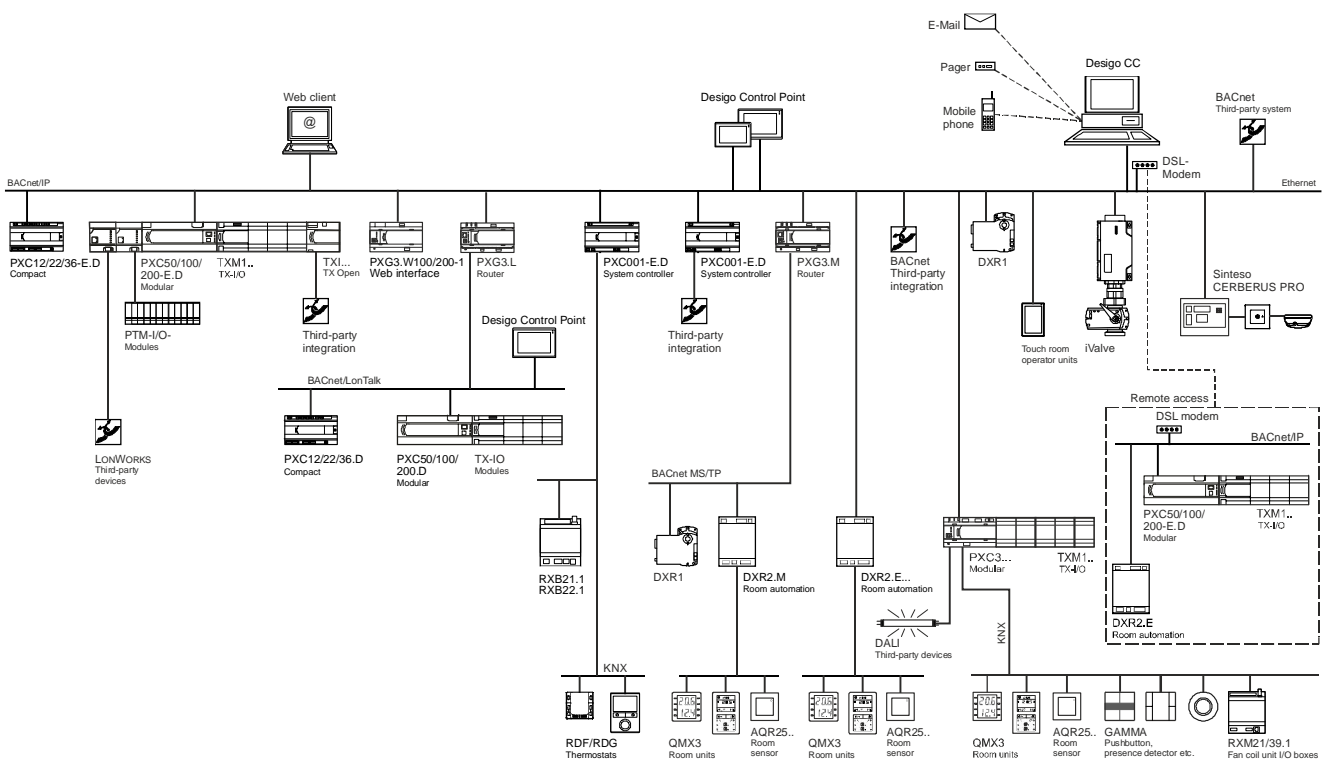
Small system on BACnet/LonTalk



Medium system



Large system



PX site

PX site is a means of structuring large PX projects. Designo room automation stations are not part of a PX site. In a PX site one PX automation station is defined as the primary server and all other PX automation stations are defined as backup servers. Every automation station can be defined as the primary server. The primary server carries out system functions, such as time synchronization, life check and the distribution of global data:

- Time synchronization: The primary server distributes the current time to the backup devices.
- Life check: The backup servers detect the failure of the primary server and the primary server detects the failure of the backup servers. If a server fails, an alarm message is sent. If the primary server fails, another automation station must be defined manually as the primary server.
- Distribution of global data: Global objects are available on all PX automation stations. The primary server synchronizes changes, e.g., calendar object, notification class object, that are made on the primary server to the backup servers.

Handling a PX Site in Desigo clients

When an operating device starts, it searches for all primary servers and offers a log in to the PX site.

A Desigo CC site can contain several PX sites and third-party devices. Desigo CC registers itself as a global alarm recipient for the PX site on the primary server.

3.8 Communication principles

Desigo uses open communications to connect various technical building systems based on open and standardized data interfaces:

- BACnet is used from room automation to the management level
- KNX®, DALI, EnOcean® and LonWorks® are used in room automation and decentralized secondary processes
- M-bus, Modbus, OPC, MS/TP, and other interfaces are used for connecting third-party devices and systems

BACnet

BACnet (Building Automation and Control Networks) is a communications protocol for building automation and control networks. BACnet ensures the interoperability between devices from different manufacturers.

See <http://en.wikipedia.org/wiki/BACnet>.

VendorID

Each BACnet device has a VendorID to identify the manufacturer. The VendorID for the Siemens BACnet system devices is 7.

BACnet over Ethernet/IP

Applications on the management level can interact via standard IT network services concurrently to BACnet services.

Desigo supports BACnet/IPv4 and BACnet/IPv6 (via PXG3.M/L router). IPv6 to IPv4 is NOT compatible. The parallel operation of IPv4 and IPv6 is possible with the use of a PXG3.L/M BACnet router.

See <http://de.wikipedia.org/wiki/IPv6>.

Network performance

The performance of the network depends on the following criteria:

- Number of devices on the bus
- Segmentation of the topology via routers (for LonTalk bus)
- Number of simultaneously active clients
- Peer-to-peer communication resulting from distributed PX applications
- Other communications services using the same transmission medium, where, e.g., office communication on a separate VLAN share the same IP trunk
- Application download on the network

Due to these factors, which can vary widely from project to project, it is not possible to make any generalized statements about network performance. If the specified product quantities are adhered to, performance is adequate.

If the network performance is not satisfactory, the following actions may help:

- Use the same automation station for items of equipment with frequent process interaction.
- Divide the network into segments via BACnet router and an Ethernet/IP backbone.
- Isolate the automation station from the network when downloading an application.

BACnet and IP network structuring

BACnet supports various application services which are transmitted to all BACnet devices (broadcasts). Global broadcasts are blocked by the IP router. BACnet solves this problem by using a BACnet Broadcast Management Device which ensures that IP broadcasts only appear in one IP segment. The logical BBMD functionality can be configured in every BACnet router and in every PX automation station with BACnet/IP. One BBMD can be configured per BACnet/IP port. Devices with BBMD must have a static IP address.

BACnet over MS/TP

MS/TP stands for Master Slave / Token Passing. Each device on the link is considered the master when it has the token. If it does not have immediate need to use the token, it passes the token along to the next device. All devices on the link which do not currently have the token are regarded as slaves, and listen to any messages the current master may have for it. As all devices take turns being master, the link is effectively peer-to-peer.

Use of other network technologies

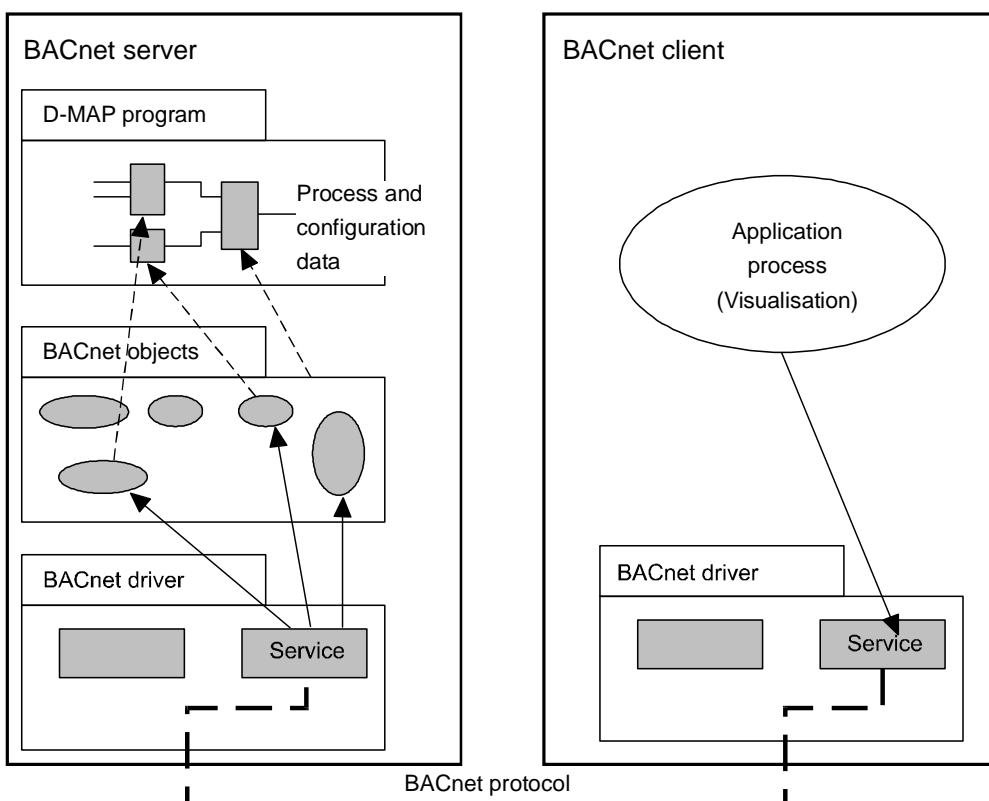
IP networks (besides the other technologies mentioned above) provide the network infrastructure Desigo devices are connected to. In case a Desigo installation is spatially distributed (e.g., several buildings on a campus, multiple branches in a country) the connection of these local IP networks (LANs) normally is done using a Wide Area Network (WAN) or a point-to-point transmission line. These can be based on non-IP technologies but typically are transparent for IP traffic. In this way, all the BACnet devices connected via an IP network can communicate with each other.

Client/Server

A BACnet device can assume two different roles within a system, the role as a server and the role as a client. These roles are defined as follows:

- Client: A system or device which uses another device via a BACnet service (service request) for a specific purpose. The client (e.g., Desigo CC, operator unit) requests a service from a server.
- Server: A system or device which responds to a given service request. The server (e.g., PXC automation station, Desigo room automation station) performs a service for a client.

Most system devices in Desigo can act either as a client or as a server, but they normally each carry out their more typical role. An automation station is normally a BACnet server, which supplies process data to other system devices. The automation station can also act as a client, when it, e.g., subscribes to a process value from another automation station.



10660Z37_01_en

BACnet standard device profile

The BACnet standard defines several device profiles that simplify to judge (and test) a device's capabilities against a specified function set. Desigo always tries to work with such profiles and prove their fulfillment by independent test laboratories and respective BTL logos and BACnet certificates.

- The PXC3 and DXR2 room automation stations comply with the B-ASC standard device profile.
- The PXC automation stations comply with the B-BC standard device profile.
- The Desigo CC management platform complies with the B-AWS standard device profile.

For a complete list with additional details, see *BACnet Protocol Implementation Conformance Statement (PICS)* (CM110665) and the products page of the BIG-EU website (www.big-eu.org).

AMEV guideline

Desigo complies with the AMEV guideline BACnet 2011 Version 1.2 with the following profiles:

- Desigo CC: AMEV profile MOU-B
- Desigo PX: AMEV profile AS-B

Desigo room automation

BACnet is used to exchange information between PX automation stations and DXR2 and PXC3 room automations stations and the management platform.

Desigo RX

The Desigo RXB room automation range communicates via KNX S-Mode (EIB).

Restrictions for LonWorks

A LonWorks network cannot be segmented with LonWorks routers, as the message length for BACnet is 228 bytes for performance reasons. Commercially available LonWorks routers do not have sufficiently large buffers for this length. No other media (power lines, infrared, etc.) can be used either.

For performance reasons, we do not recommend the operation of LonWorks and BACnet devices on the same LonTalk cable.

3.9 Data maintenance

A running Desigo system contains various categories of data, each with different requirements in terms of consistency, period of useful life and visibility. The data is distributed throughout the system, with each category having a unique origin. There is no central data maintenance in the Desigo system. The system data is distributed on all devices throughout the network, but is primarily located in the automation stations.

During the sales, planning, engineering and commissioning phases, project data is created. Part of the data is loaded into the system, while another part is tool-specific and used, e.g., for documentation of the project.

System data is:

- Process-data and parameter settings
- Archived data
- Configuration and description data
- Metadata
- D-MAP program
- Graphics and masks
- Libraries
- Offline trend object values

Process-data and parameter settings

Process data

Process data is data generated by the physical process in the building using a process control algorithm. Process data represents the process variables, such as a temperature or a damper position.

Parameter settings

Parameter settings are function parameters, settings, setpoints, etc. which are defined for each plant or project and which affect the way in which an application works. Parameter settings can be modified during operation.

Process data and parameter settings can be accessed within the system via BACnet objects, e.g., Present Value [PrVal] and Status [StaFlg], if the associated mapping is enabled in the engineering phase.

If process data is used by several automation stations, the data origin is the location where the physical variable is measured (e.g., outdoor temperature) or generated (e.g., the control signal from a time schedule). Copies are updated on an event-driven basis after a short delay.

Displaying process data and parameter settings

To display process data and parameter settings mapped to BACnet on clients, only one copy of the data needed for current operation and monitoring is stored. The Desigo system does not store complete copies of process data or parameter settings. The data (copy) required by a client is normally updated via the BACnet protocol on an event-driven basis and with a short time delay.

All process data and parameter settings, even those that are not mapped to BACnet objects (engineering setting), can be monitored and operated in Xworks Plus (XWP). BACnet clients only see what is available via BACnet.

If several clients modify the same process data, the last change is accepted.

Volatile and non-volatile process data and parameter settings

The majority of the process data is volatile data, which is recalculated when the automation stations are restarted. However, certain process data is retained even after an automation station restart, e.g., self-adaptive control parameters, run-time totalizers, etc., which are specifically identified as such in a function block. Even in the event of a program change, this non-volatile process data remains in memory and can be read back with XWP.

All parameter settings are non-volatile, that is, they are retained in the event of a power failure.

Readback

All non-volatile PX process data and parameter settings can be read back into XWP. However, parameter settings in the operator unit cannot be read back into a tool.

Global parameter settings

Some parameter settings are identical in all automation stations, e.g., date and time, calendar function blocks and Notification Class function blocks. To ensure consistency, they are held in global objects which are automatically replicated in the system.

Archived data

Setting parameters can be logged and archived. Archived data illustrates the response of process or system variables or events over a time period, e.g., trend data can be moved from the trend database into archive files. Archived data are typically lists of one or more of the aforementioned variables and are preferably stored and processed on the management level. Only small amounts of data are archived at the automation level. Such data is normally forwarded to the management level.

Ensuring consistency

Archived data only requires a consistency check in cases where it has been moved from one application to another, e.g., from the automation level to the management level. The data origin is not deleted until a check has been carried out to ensure that the data has been transferred in full. This data is stored in the non-volatile memory.

Irregularities in the logging of archived data are recorded in the data itself.

The life of the data is determined either by the user or by a configurable application which automatically condenses or deletes this archived data.

Configuration and description data

Configuration and description data is data which is defined for a specific system or project and only affects the appearance and response of the plant for operation and monitoring purposes. Some configuration parameters are tool-specific and control the options in XWP (e.g., connection allowed / not allowed, etc.). Most configuration parameters, however, are mapped to BACnet and are available to the clients. Typical data in this category is COV increment, operating limits, access level, descriptive text, engineering unit, etc.

This data is defined during engineering and always originates in the tool itself. Normally, the data is predefined with likely default values or even generated automatically from the context. This data is static and cannot be modified during operation. It is therefore not subject to consistency problems, and may be duplicated elsewhere in the system to improve performance. If engineering changes are made, you must ensure manually (through data import) that the copies are identical to the original data in the engineering tool.

This data cannot be read back from the automation stations, and must therefore be stored with the project data.

Metadata

Metadata is project-independent data from standard BACnet objects (e.g., analog input, schedule, etc.) which needs to be known by a tool or a client, e.g., texts for predefined BACnet enumeration, maximum size of arrays, data-type information, fixed operating limits, etc. The metadata is loaded into the relevant clients or tools at HQ and (except texts) cannot be modified after delivery. Text, like the text for BACnet enumeration referred to above, must be localized (language translation) and distributed to the clients and tools. This is part of the localization process.

D-MAP program

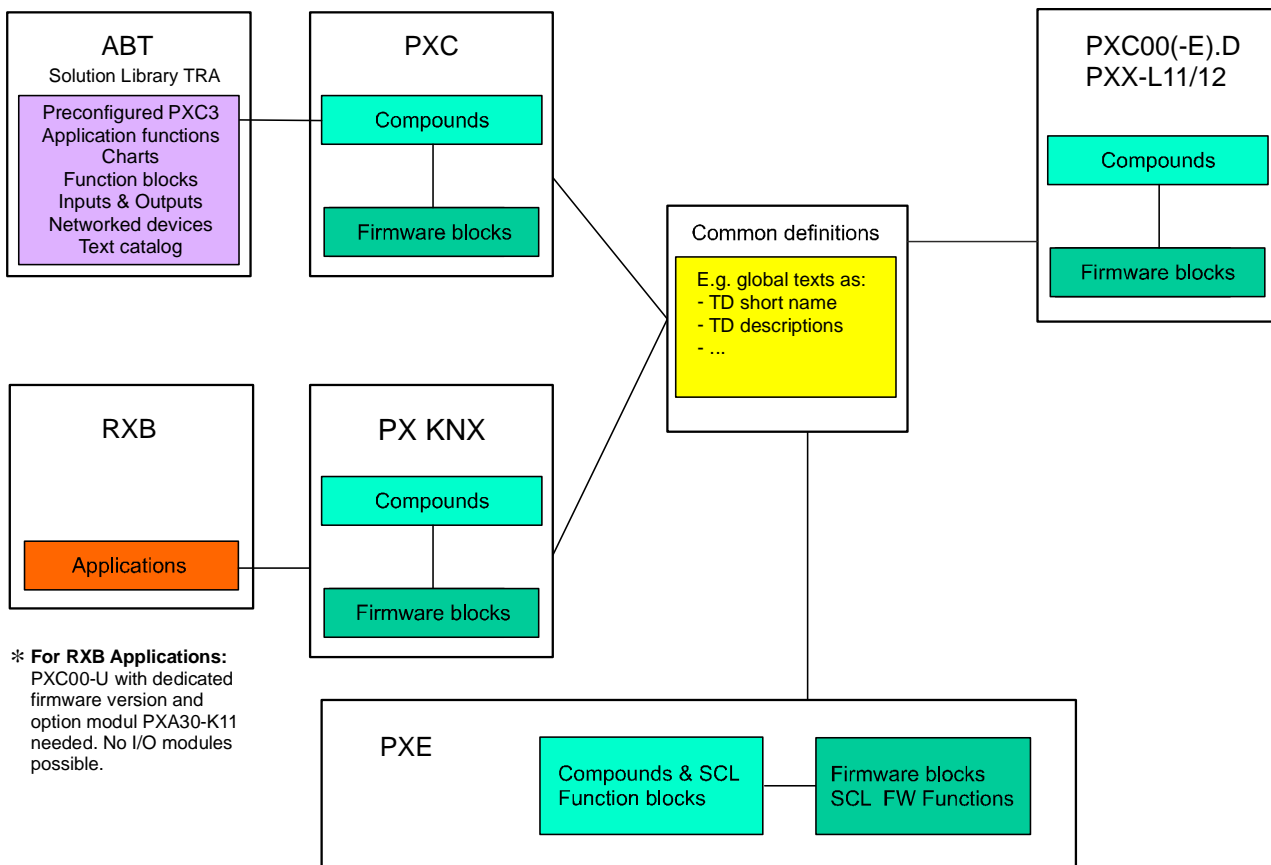
The D-MAP program is an executable program, and contains instances of the function blocks with the associated process data and parameter settings, the configuration and description data and the interconnection and order of processing of function blocks.

The D-MAP program can be modified during operation either by reloading the complete program including any changes, or by delta (differential) loading. Delta loading only reloads the changes.

The D-MAP program is generated in XWP/ABT from the information in the program charts, compiled and downloaded into the automation station.

Libraries (LibSet)

The Desigo Library Set (LibSet) is a set of mutually interdependent libraries that belong to a given Desigo system version.



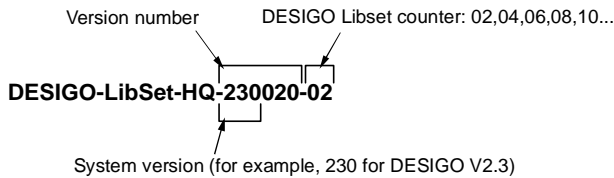
The library contents are continuously extended. Every *LibSet Extension of Desigo* (LED) is a comprehensively tested collection of solutions covering all the necessary parts of the Desigo system.

The LibSet version number defines which LED runs on which system version. The first part of the version number represents the applicable system version.

A LED includes the latest library per automation station type (PXC, PXC00(-E).D, PXX-L11/12, PXKNX) for the latest Valid Version Set.

New LEDs are delivered at regular intervals. The individual LEDs are consecutively numbered (LED0 to LED16).

LibSet version number



Desigo LibSet consists of various libraries for all system levels:

- Shared text library for PXC, PX KNX, PXE, PXR
- PXC library
- PXKNX library (RXB)
- PXE library
- PXE SCL library (Structured Control Language)
- PXR library
- Library to monitor primary plants
- Library for collaboration between Desigo PX and Desigo room automation
- ABT library (Desigo Room Automation Solution Library)

LibSet version number and LED

When a LibSet version number is released (new LED), the incremental part of the version number is increased accordingly, e.g.: Desigo-LibSet-HQ-410080-10 > Desigo-LibSet-HQ-410080-20

The remaining numerical values in the decade (e.g., 11 to 19) can be used by the RCs for localization versions. If the version number changes, the LibSet number is reset to 10 again. If the scope of the Desigo application changes, the LED number is also incremented, e.g.: LED02 > LED03

Desigo LibSet

| LED | Description | LibSet version number | Date |
|-------|--|--|----------------|
| LED00 | Basic application content for LibSet | Desigo LibSet-HQ-220031-02 | August 2003 |
| LED01 | PXC: Additional applications for ventilation and heat generation and distribution | Desigo LibSet-HQ-220031-04 | October 2003 |
| LED02 | All RXC applications for refrigeration generation and distribution | Desigo LibSet-HQ-220041-02 | December 2004 |
| LED03 | PXC: Applications for refrigeration generation and distribution RXC: Additional combined applications (INT..) | Desigo-Libset-HQ-220041-06 | March 2004 |
| LED04 | PXC: Air quality and domestic hot water applications and recovery function after power failure | Desigo-Libset-HQ-220041-08 | June 2004 |
| LED05 | RXB room automation PXC: District heating application Temperature cascade / Humid supply air control Field test version for peak load program | Desigo LibSet-HQ-230010-02 | September 2004 |
| LED06 | PXC: Additional applications for ventilation and domestic hot water Desigo Insight: Update to genie library for Visonik, Unigr and Integral | Desigo LibSet-HQ-230010-02 | January 2005 |
| LED07 | PXC: Additional solutions for ventilation facilities, refrigeration plants, heating functions, heating plants and universal functions | Desigo-Libset-HQ-230010-06 | November 2005 |
| LED08 | PXC: Like LED07 and compounds for QAX, RX DI: Genies for lab management integration PXR: Compounds for Lab Management integration | Desigo-Libset-HQ-235040-02 | November 2005 |
| LED10 | PXC: Heating degree days, three-point actuator, storage management, adjustment of humidity control | Desigo-Libset-HQ-235040-04 | July 2006 |
| LED11 | Like LED10 and RXB and RXL integration solutions | Desigo-Libset-HQ-236040-02 | July 2006 |
| LED12 | PXC: Solution for combined heating/cooling circuit, room model, quality monitoring of control circuits, leakage suppression PX/KNX: New integration compounds | Desigo-Libset-HQ-237030-02 Desigo-LibSet-HQ-236050-04 | February 2007 |
| LED13 | PX Open compounds | Desigo-LibSet-HQ-237070-02 | December 2008 |
| LED14 | PXC: Additional applications for ventilation facilities, heating/refrigeration circuit, heating circuit Heat storage tank and trend | Desigo-LibSet-HQ-400210-10 | March 2009 |

| LED | Description | LibSet version number | Date |
|--------|---|----------------------------------|--------------|
| LED15 | PXC: Energy-efficient application AirOptiControl for ventilation and air conditioning plants Compounds to integrate Grundfos and Wilo pumps | Desigo-Libset-HQ-410090-10 | April 2010 |
| LED16 | PXC: CAS21 (HVAC) Compound for Desigo room automation demand signals, compounds for pumps and fans based on PTM16.xx PXC: CRS01 (Collaboration Room Solutions) Compounds for Desigo room automation collaboration PXC: MON01 (Eco monitoring) Monitoring compounds und standard solutions for monitoring primary plants ABT (Desigo room automation): - TRA02_V5.0_HQ_ABT1.0 (for firmware TRA V5.0)* - TRA03_V5.0_HQ_ABT1.0 (for firmware TRA V5.1)* Basic library for integrated Desigo room automation room solutions (HVAC/Lighting/Shading) -TRA01_QMX3V5.0_V5.1_HQ_ABT1.1 (firmware TRA V5.1)* Like TRA02/TRA03_V5.0_HQ_ABT1.0 (see above) plus room units QMX3.P34, QMX3.P34, QMX3.P37, QMX3.P02 with V5.0 functionality like with QMX3.P36 | Desigo-Libset-HQ-500204-10 | March 2012 |
| LED17 | PXC: CAS22 (HVAC) Integration variable speed drive G120P | Desigo-Libset-HQ-500260-10 | October 2012 |
| LED20 | PXC: Ventilation & air conditioning: Extensions for night ventilation, room temperature monitoring, predefined trend objects, timer function, temperature and humidity control, outside temperature controlled heating and cooling function, fire control Heating: Extensions for hot water coordinator. PX KNX: CAS09 Integration RDG/RDF/RDG ABT (Desigo room automation): - TRA01_V5.1_HQ_ABT1.1 (for firmware TRA V5.1)* VAV application extensions, chilled ceiling and fan coil application, boost heating and optimum start/stop, air quality applications, extended support for QMX3/AQR25 | Desigo-Libset-HQ-510xxx-10 | Summer 2013 |
| LED21 | PXC: - MON Library changed Compounds: SetRib-Pin at KPI is set, if there is invalid information - This state is displayed in EcoViewer, All Observers are now changed to reduced I/Os ABT: - TRA03_V5.1SP_HQ_ABT1.1 - Extended support for QMX7 | Desigo-Libset-HQ-51SPx-10 | March 2014 |
| LED 22 | PXC: CAS26 - New: Compounds for integration of Variable Speed Drive G120P MSTP PX KNX: CAS10 - New: Compounds for integration of RDG100, -160, -165 / RDF301, -600, 800 / RDD-810 | Desigo_Libset_LED21-HQ-510212-10 | May 15 |

| LED | Description | LibSet version number | Date |
|---------|---|----------------------------------|-------------|
| LED 23 | <p>PXC: CAS26</p> <ul style="list-style-type: none"> - New: Compounds for integration of Variable Speed Drive G120P MSTP - New/changed: Compounds with supply coordination functions to support collaboration of Desigo Room Automation with Desigo PX <p>PX KNX: CAS10</p> <ul style="list-style-type: none"> - New: Compounds for integration of RDG100, -160, -165 / RDF301, -600, 800 / RDD-810 <p>PXC: CRS03</p> <ul style="list-style-type: none"> - New/changed: Compounds with supervisory functions to support collaboration of Desigo Room Automation with Desigo PX <p>ABT</p> <ul style="list-style-type: none"> - ABT Site Library with BACnet IP devices (ABT Site01_V6.0_HQ_ABT1.2 MSTP) - ABT Site Library with BACnet MS/TP devices (ABT Site01_V6.0_HQ_ABT1.2 IP) - ABT Pro Library with SI engineering unit (ABT Pro01_V6.0_HQ_ABT1.2_SI) - ABT Pro Library with US engineering unit (ABT Pro01_V6.0_HQ_ABT1.2_US) - ABT Pro Library with Imperial engineering unit (ABT Pro01_V6.0_HQ_ABT1.2_IM) - ABT Pro Library with Canadian engineering unit (ABT Pro01_V6.0_HQ_ABT1.2_CA) | Desigo-Libset_LED23-HQ-600xxx-xx | August 15 |
| LED 24 | <p>PXC: CAS27</p> <p>PXC: MON02</p> <p>PXC: CRS03</p> <p>PXKNX: CAS10</p> <p>New: Includes enhancements to the existing CAS library & additional functionality, e.g.:</p> <ul style="list-style-type: none"> - Harmonization of CFC charts for better usability - All plants are equipped now with trends - 4 new burner & 5 new boiler compounds and a multi-boiler plant with up to 8 boilers - New logics in heating & cooling plants for power up/down - Heating distribution for direct room control - Improved alarming concept - Improved plant behavior on sensor disturbance (heating distribution and domestic hot water) - Defects solved | Desigo-Libset_LED24-HQ-610172-10 | July 17 |
| NEW LED | New: Includes new compound {IngtVlv1} for iValve in the PXC library | Fits with Desigo V6.2 Update | Summer 2019 |
| NEW LED | <p>PXC: CRS04</p> <p>New: Includes new compound {HcLgtPrf1} for Human-Centric Lighting in the PXC library</p> | Desigo V6.3 | Summer 2021 |

Key

- * The PXC3 room automation station supports several firmware versions independent of the functional content of the application library.

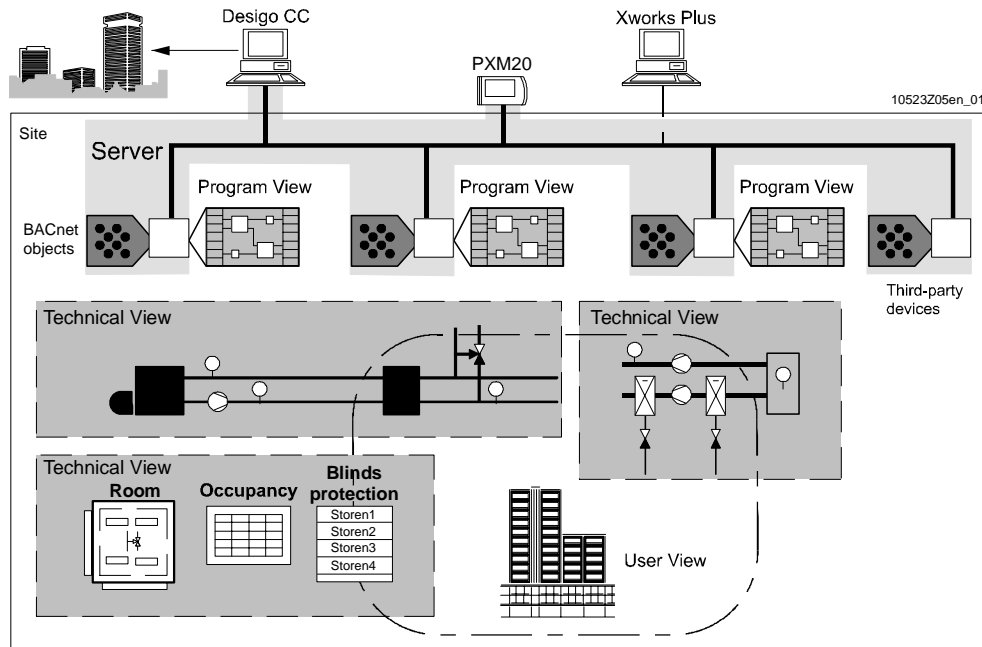
Desigo CC

The application libraries for Desigo CC are delivered as extension modules for the respective system versions. For information about compatibility, see *Desigo CC System Description (A6V10415500)*.

3.10 Views

There are four views:

- Technical view
- User view
- System view
- Program view



Technical view

The technical view illustrates the technical building services equipment, such as HVAC systems and associated elements, in the building automation and control system. The technical view is always present and can be used as a substitute for the user view if the user does not have his own user designation.

User view

Freely defined and structural user view

The user view is optional in a project. The user view is based on user designations, e.g.:
PL7'FL3'ELE"HEAT.STPT

The structure and syntax of the user designations can be defined for each specific project and customer.
Example of a structure: Installation/building/room/plant element/signal

User view via the User Designation (UD)

Desigo supports different user views, depending on the application:

In Xworks Plus (XWP) a User Designation (UD) can be entered for function blocks or compounds in addition to the Technical Designation (TD) and description. This entry is carried through in the system and can be evaluated by clients. The UD allows customers to use their own preferred designations for the plant without changing the technical structure. The UD can be used in the management platform in addition to the TD. The detailed view in the PXM20 operator units shows the UD as information.

User Designation for Desigo room automation

You can define the user view for Desigo room automation as follows:

- Define a structure for the user view
- Copy Desigo room automation objects into the user view
- Define UD's that can be used as object names

System view

The system view shows the standard system hierarchy (BACnet view):

- Network, topology
- Device and third-party device view
- Flat representation (no hierarchy) of all BACnet objects in one device

The system view provides access to all BACnet devices (including third-party BACnet devices) and all BACnet objects. A third-party client displays this view of a PX device.

The system view is used in the PXM20 only for third-party devices.

Program view

The engineering and program view corresponds to the XWP/ABT view. The structure is matched to the automation station. Within an automation station, the view is program oriented: nested CFC charts (compounds) and function block instances.

Views and users

The views reflect the differing needs of their users.

| Per | User | Technical view | User view | System view | Program view |
|-----|--|----------------|-----------|--------------|--------------|
| 1 | Operator (without technical training) | Main view | Main view | No access | No access |
| 2 | Operator (with technical training) | Main view | Main view | Occasionally | No access |
| 3 | Engineer (Desigo CC), User (PXM...) | Main view | Main view | Occasionally | No access |
| 4 | Service engineer, Siemens service engineer | Main view | Rarely | Rarely | Main view |

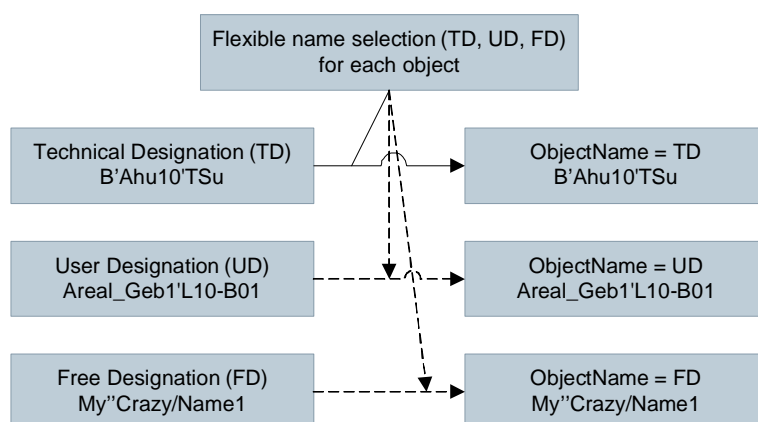
Flexible object name / device ID engineering

You can flexibly generate the object name during engineering in XWP. This is called the Free Designation (FD). However, the FD has no inherent hierarchical structure, which makes it tedious to engineer and lowers its helpfulness to orientate in larger buildings. It should thus be considered as a naming type for very special purposes only.

Flexible object name engineering causes a greater engineering effort and must thus be requested specifically by the customer.

Each BACnet object has an object name for identification on the BACnet network. This object name must be unique within the automation station. The Technical Designation (TD) is used as default for the object names. The TD is a technical identifier and is used to identify the plant and associated elements in the technical view.

You can select how the object name is created for each standard BACnet object. This especially applies to BACnet multivendor projects, where a special object name structure is required.



Defaults and rules

The following defaults and rules apply when you engineer the object name in the XWP Hierarchy Viewer:

- The Free Designation (FD) can be max.69 characters.
- Only ISO-Latin-1 code points from [32..127] and [160..255] may be used. This excludes all characters from [0..31] and [128..159]. These ISO-Latin-1 code points are identical to Unicode code points.
- No lead or post blanks [32] may exist and object names containing only blank characters are not possible.

The FD values and the object name selection are transferred automatically to the automation station or exported to the management platform during compiling or loading in the CFC.

The CFC Editor checks during compilation if the object name is unique for each automation station under the following rules:

- The same resulting object name may exist only once per automation station. This also applies to the device object that must be unique in the BACnet internetwork.
- The resulting object name may not correspond to a TD of another object in a Device. The TD is used to resolve BACnet references.

Exceptions for object name assignment

Object names cannot be engineered in CFC charts or compounds. These elements always define the TD and the object name is always the same as the TD.

Special blocks, such as Heatcurve and Discipline I/Os generate reduced value objects in the background whose object name per default is the TD during compilation.

Free definition of the Device ID

The device ID (the object ID of the device object) can be freely defined. Range: 0...4'194'303

4 Desigo workflow, tools and programming

The Desigo tools cover parts of the technical process and parts of the Desigo system.

Main tools

The most important tools are:

- Desigo Configuration Module (DCM): For designing the Desigo system in the sales phase
- Xworks Plus (XWP): For engineering, commissioning and servicing Desigo PX system components
- Automation Building Tool (ABT): For engineering, commissioning and servicing Desigo system components

Special tools

There are also special tools, e.g.:

- Tools for configuring and commissioning specific product families, such as RXT10 for the configuration of room devices on LON
- Tools for specific tasks, such as the AL Migration Tool for the migration of legacy system components to Desigo PX

See *Automation level migration, Engineering manual (CM110776)*.

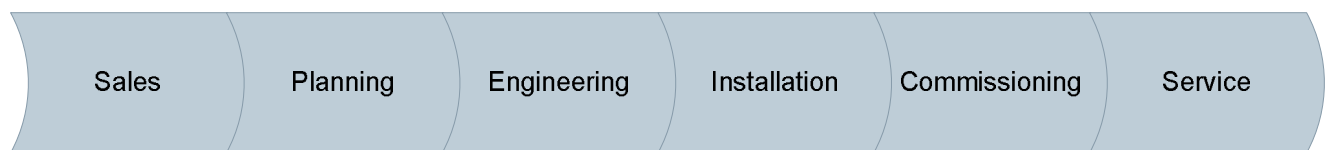
4.1 Coverage of the technical process

The Desigo tools are used in the technical process, especially for designing the system in the sales process, for engineering, commissioning, and servicing. The tools have interfaces to specific tools of the regional companies, such as tools for designing electrical wiring diagrams.

The Desigo tools cover the entire technical process from sales to service:

- Sales
- Planning
- Engineering
- Installation
- Commissioning
- Service

For service operations the Desigo tools support remote data access to project data via Branch Office Server (BOS). SSP provides the service platform.



Coverage of the technical process by Desigo tools:

| Europe | Sales | Planning | Engineering | Installation | Commissioning | Service |
|--------|-------|----------|-------------|--------------|---------------|---------|
| STST | • | | | | | |
| DCM | • | | | | | |
| XWP | | | • | • | • | • |
| ABT | | | • | • | • | • |

Coverage of the technical process by Apogee and Desigo Tools:

| USA | Sales | Planning | Engineering | Installation | Commissioning | Service |
|-----------------|-------|----------|-------------|--------------|---------------|---------|
| STST | • | | | | | |
| DCM | • | | | | | |
| ABT | | | • | • | • | • |
| Apogee tools | | | • | • | • | • |
| Desigo tool set | • | • | • | | | |

Sales

DCM supports system design and quantity determination during the sales process. Price calculation, offer preparation and tracking, and invitation of tenders are supported by country-specific tools.

Planning

The planning tools are country-specific and comprise the following:

- Network planning, design and documentation
- Cable planning and design (network cables, field device cables)
- Texts for equipment plates
- Building planning (system components in the building, room segmentation)
- Plant planning and documentation (plant schematics, function description)
- Planning of the groupings for Desigo room automation
- Order lists

Engineering

Most of the Desigo system components are engineered offline, before they are commissioned. This way you can verify and document the configuration (e.g., for the uniqueness of addresses), and define work packages for subcontracting.

XWP and ABT are Desigo engineering tools and allow the following:

- Engineering the primary equipment, room automation, BACnet router
- BACnet references for the integration from/to third party systems
- Interfaces to ElektroCAD, Pharma Validation
- Exports for documentation
- Export for engineering in Desigo CC

The tool export:

- Generates information for illustrating the generic operation (technical hierarchy, User Designation hierarchy)
- Contains information for efficiently generating graphics (mapping functions to symbols and graphic templates in Desigo CC).

Installation

XWP and ABT allow the following:

- Creating order lists that can be used for ordering the devices
- CAD export for connecting to ElektroCAD for designing control cabinets
- Parallel working of several subcontractors/engineers in a project
- Creating pack and go's for commissioning and the point test for subcontractors
- Loading configurations
- Creating commissioning data point lists

Commissioning

XWP and ABT allow the following:

- Commissioning of the systems (loading programs, program function test)
- Online trending during commissioning

- Diagnostics during commissioning
- Parallel working of several commissioning engineers in the project

Service

XWP and ABT allow the following:

- Data access to Branch Office Server (central engineering data management of the regional companies)
- Data security (reading system data in the engineering database)
- Remote engineering and operating, diagnostics and error recovery via an external network connection

4.2 Coverage of the system

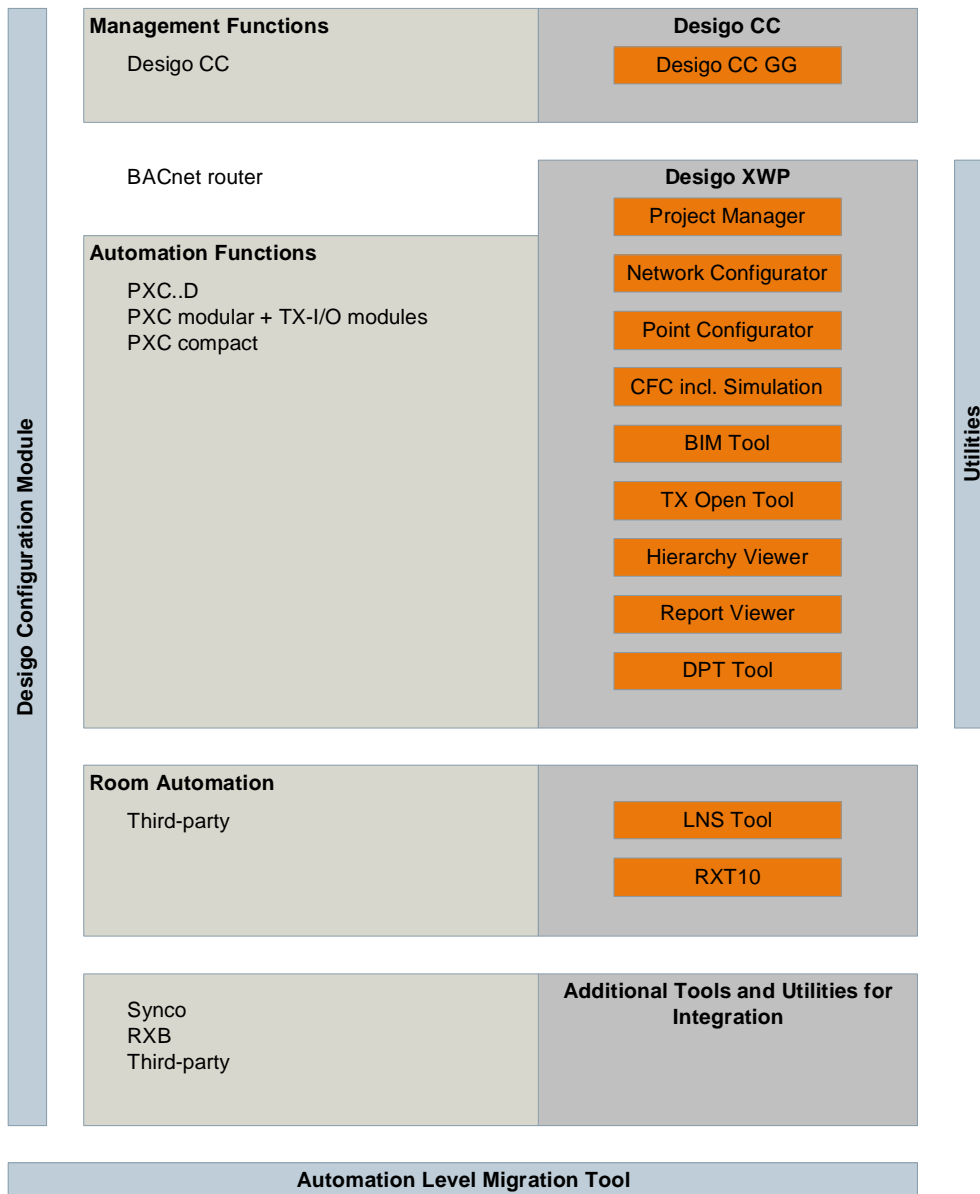
The Desigo tools cover all levels of the Desigo system except the management level:

- Xworks Plus (XWP) covers Desigo PX.
- Automation Building Tool (ABT) covers Desigo primary plants and room automation.

Tools for Desigo PX

The following tools are used for Desigo PX:

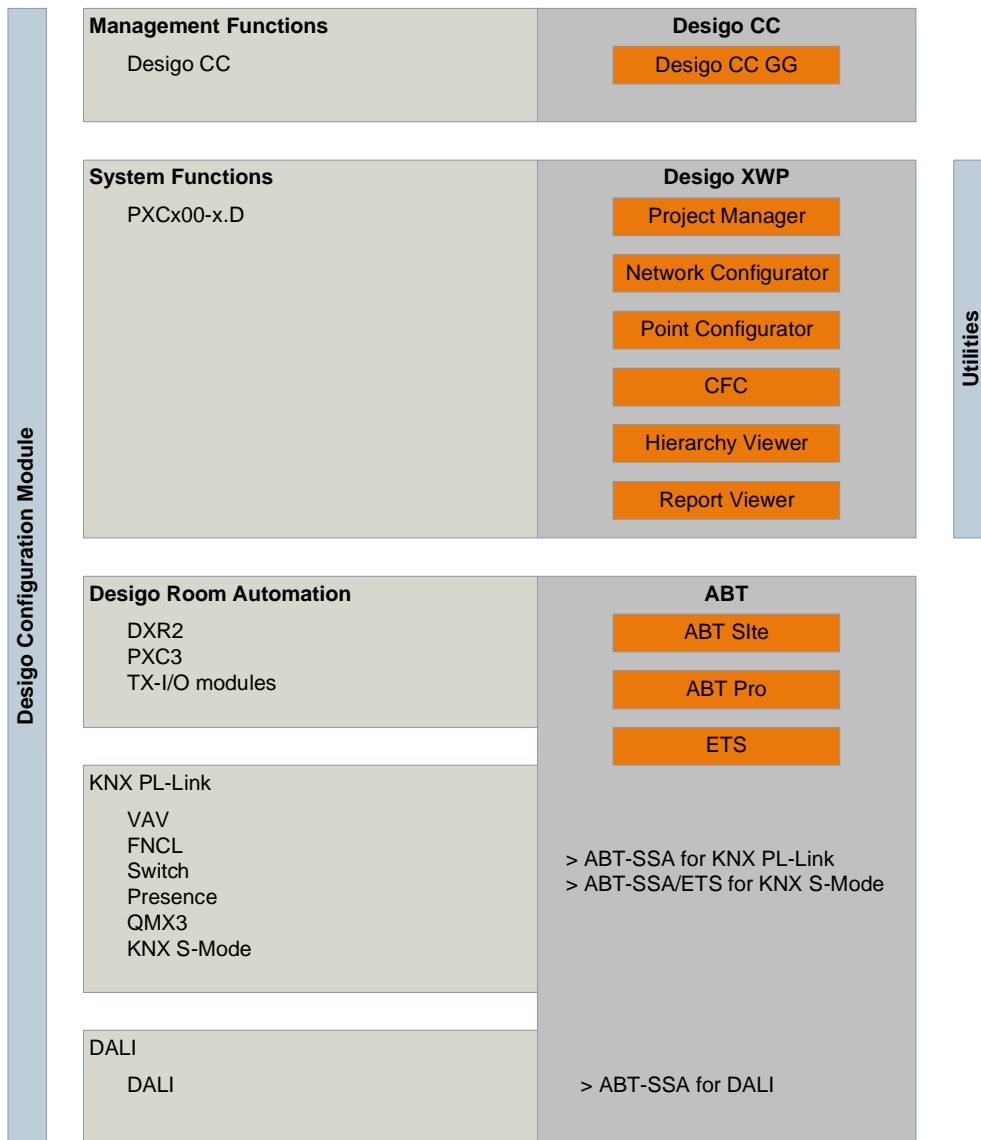
- DCM: For designing the system and determining the necessary quantities
- XWP: For configuring and commissioning BACnet routers
- ACS: For configuring, commissioning and operating Synco and RXB devices
- PX KNX Tool: For configuring the KNX side of the PX KNX system controller
- AL Migration Tools: For migrating Unigyr, Visonik and Integral to Desigo



Tools for Desigo primary plants and room automation

The following tools are used for Desigo primary plants and room automation:

- DCM: For designing the system and determining the necessary quantities
- XWP/ABT:
 - For configuring, programming and loading PXC../DXR2 automation stations
 - For integrating KNX devices into Desigo room automation (on KNX PL Link Bus)
 - For engineering and commissioning PXC3, TX-IO, In-Room Bus DALI and KNX PL Link



4.3 Main tasks

What's covered by the Desigo tools?

The Desigo tools let you design, document and maintain Desigo systems, that is, you design and document technical configurations and programs for the Desigo system.

What's not covered by the Desigo tools?

The following processes and products are covered locally by SSP or VAPs and not by the Desigo tools:

- Sales: Offer preparation and tracking
- Planning/Engineering: Network planning and design, floor plan, cabling, designing control cabinets, designing electrical wiring diagrams, creating rating plates, validating pharma systems
- Project management: Ordering devices, project planning, claim management, project task planning
- Service management: Service database for devices, network planning, remote service platform

Sales support

Desigo Configuration Module (DCM) supports the calculation of the Desigo configuration for the sales process. You can verify if:

- The Desigo configuration is technically correct, that is, the solution that was sold can be implemented with Desigo
- The system limits have been taken into account, that is, the number of possible devices and functions in the network is verified
- The quantity is correct, that is, correct device types for the automation and room functions, field devices, accessories and licenses
- Services are correctly calculated
- The design for the review with the customer is well documented
- Prices are correct (regional companies can add their prices to the DCM database)

Configuration and programming

The configuration and programming flexibility of system devices depends on the product or product family. Some devices contain preloaded applications and connections only to specific periphery device types.

You can configure and parameterize the devices offline or partly online with a configuration tool. You can replace the preloaded applications on some devices in the project.

You can freely program some devices. To create loadable applications, you can use libraries to assemble project-specific solutions.

Degree of standardization and flexibility

The table below shows engineering methods by degree of standardization and flexibility:

- Level A: High degree of standardization with predetermined flexibility
- ...
- Level E: Low degree of standardization with very high flexibility

Solutions with a high degree of standardization

Solutions with a high degree of standardization are:

- More efficient to configure and commission than freely programmed solutions
- Easier to maintain, because the functions are verified and well documented

Solutions with a low degree of standardization

Solutions with a low degree of standardization, that is, freely programmed solutions, are:

- More laborious to create and document
- More error-prone than verified solutions
- Harder to maintain in the service phase, because they do not adhere to the standard and are often not as well documented as verified solutions

The intermediate levels B, C and D allow you to choose a solution with the right balance of flexibility and standardization.

Desigo PX:

| | Level | Description | Library | Example | Engineering effort |
|------------------|-------|---|---|-------------------------|--------------------|
| Standard | A | Solution Browser in XWP | Locked CAS solutions | AHU10 | Low |
| | B | Solution Configurator in CFC, CAS library | CAS solutions, aggregates, components | AHU10, fan, valve | |
| | C | CFC programming, CFC library | Charts, blocks | CAS library with charts | |
| High flexibility | D | CFC programming, solution creation | Charts, blocks, LMU (Library Maintenance Utility), simulation | RC library | High |
| | E | CFC programming, SCL block creation | CFC, SCL, simulation, LMU, development tools | HQ library | |

Desigo room automation:

| | Level | Description | Library | Example | Engineering effort |
|------------------|-------|-------------------------|---------------------|------------------------|--------------------|
| Standard | A | Application selection | Application type | Standard, VAV | Low |
| | B | Application assembling | Application modules | Blind, radiator, light | |
| | C | Application creation | XFBs | VAV | |
| High flexibility | D | Application engineering | CFC FB's | Regional specialties | High |
| | E | Development | All levels | All | |

Level A

You can create solutions with the available options and variants with little prior training and detailed knowledge. The device is preconfigured and can be configured for the specific project. The functions are predefined. You can configure the application using options and variants. You can set the function of the application and the peripheral devices with a configuration tool. The solutions are delivered by HQ as verified and documented solutions.

Level B

The device can be configured for the specific project. You can assemble the application using library elements. This is a major advantage of the Desigo application libraries. Even though assembling a solution is relatively easy, the functions of the solutions are powerful. Using many options and variants, you can customize the standard solutions to your project requirements.

Level C

The device is preconfigured and can be configured for the specific project. You can assemble the application using library elements. You can program the application with default function modules with predefined interfaces. You can program using simple programming functions.

Level D

This level offers full flexibility, but requires detailed knowledge of the application's structure, the programming tools, BACnet and the Desigo system functions. You can program in CFC (Continuous Function Chart) with basic function modules. You can use all available programming functions. You must ensure that the programs you develop fit together regarding execution, priorities, auto-connecting in the tool, interface usage, etc.

Level E

This level offers full flexibility, but requires detailed knowledge of the application's structure and the programming tools. You must ensure that the functions of the program work. You must ensure that the programs you develop fit together with all elements in the library and that they are well tested and documented. You must take care of the compatibility, the versioning and the library packaging.

Creating a technical hierarchy

The technical hierarchy is the BACnet view on the Desigo system. It is based on the plant-related structure in the building. This hierarchy is defined during engineering. In special cases, if the customer requires it, the technical hierarchy can be built according to a plant-specific structure defined by the customer (user designation).

This lets, e.g., the customer view the building in Desigo CC according to this structure:

- Building topology (area, building, floor, plant, plant section, etc.)
- Naming in the system (names according to technical hierarchy, user designation or free designation)

Creating loadable components for the automation stations

The result of the engineering are loadable configurations:

- Configuration of the automation station: Network configuration (IP, LON, MS TP addresses), BACnet configuration (BACnet name and BACnet ID)
- Application: I/O configuration and setting parameters or program (for programmable automation stations)

- Operating language: When you load the configuration, the operating language for the generic operation is also loaded
- Firmware: For system upgrade or bug fixing

Creating the configuration of operation

The system devices can be operated locally, over the web, on a touch panel or in Desigo CC. Operations can either be generic (without additional engineering) or dedicated (with additional engineering via favorites or operating graphics).

- The generic operation is based on the technical hierarchy. It must not be engineered.
- The room operation can be configured.
- Favorites are a simple grouping of operable elements in a summarized view. This view can also be generic, e.g., as a favorite in ABT-SSA, or it can be engineered, e.g., as a favorite for PXM20.
- The graphic operation must be engineered.

Installation, test and commissioning

An I/O configuration must be loaded for the point test. An application program is not always loaded with the configuration.

You can carry out a point test with an application program if the application program can be turned off during the point test. This way you can carry out a test if, e.g., a central security function would prohibit you from operating the I/O, e.g., if a central security function does not allow lowering the blinds.

The test protocol can store which points have passed the test and which points have an error.

Creating local documentation and project documentation

The tools have two types of documentation:

- Local documents (work documents, simple templates, Excel exports) can be used to, e.g., verify results. You can, e.g., export them to Excel and add additional data to them.
- Project documentation (template with logo, author, table of contents, etc.) can be attached to the customer documentation either in printed form or as a PDF.

Managing project data

You can manage project data in three ways:

- Local project data management - You can save project data locally, that is, on the local computer or on a share.
- Project data backup - You can create project data archives to, e.g., locally save the intermediate status of engineering data.
- Project data on the Branch Office Server (BOS) - You can store project data on a BOS. This allows:
 - Data storage on a server, incl. data backup
 - Control of project data access, through login data
 - Checking project parts in and out for working on engineering data in parallel

4.4 Tools for different roles

In a project different roles are responsible for different tasks. Based on these roles, there are various tool packages with different functions and licenses.

Roles:

| Role | Description | Application area |
|------------------------|--|----------------------------|
| Application Engineer | Can reprogram applications on a project-specific basis. | System and room automation |
| Design Engineer | Can carry out a project. Can select and configure solutions from the library. | System and room automation |
| Commissioning Engineer | Can commission solutions, Can configure applications online. | System and room automation |
| OEM, Installer | Can carry out a project. Can select, configure and commission solutions from the library. | Room automation |
| Electrical Installer | Can load configurations. Can configure devices. Can test points. | System and room automation |
| Balancer | Can balance rooms regarding air and water supply. | Room automation |

Tools for different roles:

| Tool | Tasks | Application / Design / Commissioning Engineer | Adv. OEM Installer | OEM, Installer | Electrical Installer | Balancer |
|--------------------------------------|--|---|--------------------|----------------|-----------------------|-----------------------|
| XWP | Create projects and reports, design networks, BOS | • | | | | |
| ABT Site > Projects | Create and open projects | | • | • | • | • |
| ABT Site > Building | Create building structure and grouping hierarchy | • | • | • | | |
| ABT Site > Configuration | Configure applications, mass create devices | • | • | • | | |
| ABT Pro | Configure HW, edit applications, CFC (TIA), debug | • | • | | | |
| ABT Site > Startup | Device discovery, persistence of readbacks, set up nodes, load web pages, ABT-SSA for KNX PL-Link, MS/TP | • | • | • | • | • Simple GUI |
| ABT Package | | XWP ABT | ABT | ABT Site | ABT Site not licensed | ABT Site not licensed |
| ABT-SSA Access via role and password | Point Test Operate and monitor Balancer | • • • | • • • | • • • | • • | • |
| ABT-SSA | | | | | | |

4.5 Working with libraries

Libraries ensure efficiency and quality.

HQ libraries

There are HQ libraries for every engineering level. HQ libraries:

- Allow you to work efficiently
- Are verified
- Are well documented
- Are based on a text data basis that allows you to switch the language in engineering, that is, the library is language neutral

- Are versioned
- Can be installed with the library setup

RC libraries

Based on HQ libraries, you can create country-specific RC libraries that cover country-specific function requirements.

Project-specific libraries

Project-specific libraries are based on HQ or RC libraries and contain components with the specific settings needed in the project. This lets you use reuse already configured solutions in the project.

4.6 Working in parallel and subcontracting

Project data management during parallel working

Project data management for the Desigo tools allows several users to work in parallel in different phases of the customer project, e.g.:

- Several users are engineering and commissioning in the same project
- Parts of the project are outsourced to subcontractors, e.g., for the point test

To ensure the consistency of the project data, parts of the project data are stored on the Branch Office Server (BOS). This way several engineers cannot modify the same data elements of the data basis at the same time.

Check-in/Check-out mechanism

The check-in/check-out mechanism ensures that when several users are working in parallel during engineering, commissioning or service, they cannot make changes to the same automation station. This way no inconsistent data can be created.

To quickly transfer project data, the data is compressed before it is sent from the computer to the server. The data is managed on the Branch Office Server. The project creator transfers the data from his local hard disk to the server.

In large projects the data can be moved in two steps:

1. Step: Part of the project is transferred from the Branch Office Server to a computer in the plant.
2. Step: Parts of the transferred project can be transferred to local computers. This is called a sequential check-out.

Parts of the project, such as the building or network topology are checked out in read-only mode, so that all users always have the project overview.

Working in parallel during engineering

Several users can work on different automation stations in the same project at the same time. To do this, data is transferred from the central data storage on the Branch Office Server to the local hard disks, e.g., individual automation stations are being commissioned at the customer's site while some automation stations are still being engineered at the office.

Working in parallel during commissioning

Several users can work on different automation stations in the same project at the plant at the same time. To do this, the components to be loaded are transferred (Pack & Go), so that the user, e.g., can load the configuration or program and then perform the point test. The test results are saved in the automation station and can be viewed and transferred back to the engineering database by the commissioning engineer at any time.

Working in parallel during service

A service technician can connect with the plant by remote and make changes. To do this, data is transferred from the Branch Office Server to the local hard disk. After the technician is done, the changes are transferred back to the Branch Office Server, so that the project database is up-to-date again.

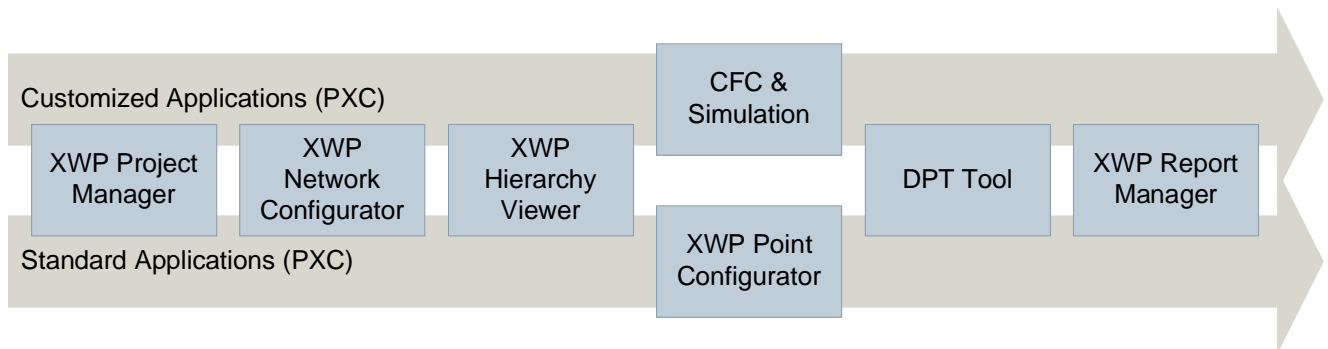
Subcontracting

Project-specific solutions can be developed outside the project organisation and specific tasks, such as configuration and point test can be outsourced to subcontractors.

If you outsource specific tasks, make sure that:

- The work packages for the subcontracting can be easily transferred to the subcontractor
- The subcontractor's work can be documented
- The changed data can be transferred back to the engineering database

4.7 Workflow for primary systems



XWP Project Manager

Create project:

- Create project
- Check project in on Branch Office Server (BOS) and define access to project
- Define project defaults
- Create control cabinet topology (local specification of the automation station, e.g., control cabinet view)

XWP Hierarchy Viewer and XWP Network Configurator

Create project structure (the building structure is system-oriented):

- Create project structure
- Create building topology (building, building parts, etc.)
- Create system topology (sites)
- Create network topology (XWP Network Configurator, third party devices, router, computer)
- BACnet references from third party devices and between primary system and room (demand signals, supervisory)
- ACP (passwords for accessing the automation stations)

XWP Point Configurator

Create systems:

- Define systems (systems, system sections, components, data points) (solutions, data points, I/O modules)
- Configure the operations (XWP Hierarchy Viewer)
 - configure the generic operation
 - Configure the project-specific operation (favorites)

CFC & Simulation

Program and configure:

- Program in CFC
- Define points in the I/O Address Editor
- Parameterize in the Parameter Editor
- Define alarming and trending

DNT and DPT

Test and commission:

- Export data to Desigo CC
- Download firmware (upgrade if necessary)
- Load configurations and programs
- Carry out point test
- Debug in CFC (if necessary)
- Create commissioning documentation (local reports)
- Specialties:
 - Integration (TX Open Tool, BIM Tool)
 - AL Mig (AL Mig Tool)
 - Simulation

XWP Report Manager

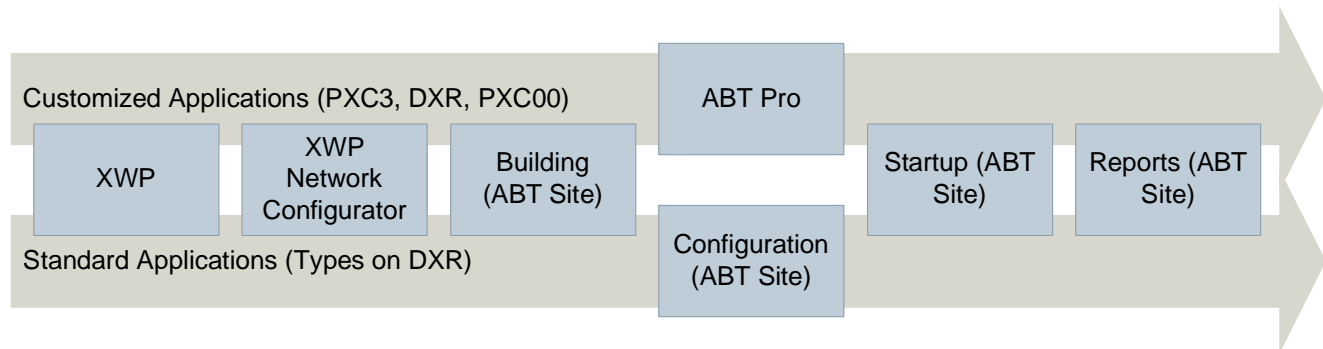
Create documentation:

- Create project documentation

4.8 Workflow for room automation classic

See *Desigo Xworks Plus Overview of Workflows* (CM111000).

4.9 Workflow for Desigo room automation



XWP

Create project:

- Create project
- ACP (passwords for accessing the automation stations)

ABT Site > Building

Create project structure (the building structure is room-oriented):

- Create building topology (buildings, floors)
- (Optional) Create user-specific building topology (UD structure)
- Create network topology (define address ranges)
- Create documentation (XWP Report Manager)

ABT Pro and ABT Site > Configuration

Create project library:

- Program automation stations (ABT Pro)
- Create templates for type-based automation stations (ABT Site > Configuration)
- Create templates for room control units

ABT Site > Building

Create instances in the building:

- Create automation stations, or rooms, based on the project-specific library per floor
- Edit room parameters

ABT Site > Startup

Commission:

- Configure and load automation stations (node setup)
- Carry out point test (ABT-SSA) (subcontracting)
- Parameterize (ABT-SSA)

All projects are password-protected.

4.10 Desigo Configuration Module (DCM)

Desigo Configuration Module (DCM) lets users, who work in sales or project execution, design the building automation and control system.

See *Desigo Configuration Module* (CM110752).

DCM is automatically updated with the latest data if installed correctly by using the provided setup program, keeping the suggested installation path, and if an online/network connection is available. The updated data can also be updated regionally or dependent on a user to reflect relevant requirements.

Field of application

DCM calculates the required materials for an installation from raw system data, such as data points, panels, and building and plant structures.

You can use DCM to conduct analysis of variants after defining and completing the installation structure by generating copies and then subsequently changing the hardware specifications. If prices are stored in DCM, you can also compare prices to find the best possible device for the money. You can copy the devices calculated in DCM from the price lists or export them as an Excel file to calculate a bid.

Flexibility

You can enter the data directly into DCM or import it as an Excel file for the automation and Desigo room automation level.

The structure in DCM is hierarchical, but you can customize the structure according to your requirements.

Management level

The required software licenses for the selected functions, devices, integrations and data points are calculated on the management level. The licenses are listed and the required software units are calculated.

Calculations can be made for new installations and for upgrades and migration. To calculate upgrades and migration, you can import existing license keys. The import provides the exact installed basis and explicitly allows additional, required licenses and software units.

Devices for the Desigo Web Interface are also determined on the management level. The calculation is based on the number of data points to be integrated in the web interface and by the number of desired touch panels.

Desigo room automation level

The Desigo room automation level lets you create highly complex building structures with the sublevels building, floor, zone, room, and room segment for Desigo room automation.

The required hardware is calculated from the specified functions and/or data points and/or KNX PL-Link and KNX devices. The specification follows a model function set that is then assigned to the structure within the Desigo room automation level. In addition, multiple model function sets can be created in a project and each of them assigned as needed. A structure at the Desigo room automation level may have multiple, assigned model function sets, and a model function set can be assigned to different structures. As Desigo room automation often uses the same structures and functions, you can indicate a multiplier on each sublevel within the Desigo room automation level.

Automation level

At the automation level you can calculate the required hardware based on specified data points.

You can choose and calculate many variants using presets. Variants include, e.g., the automation station type or I/O module type, larger automation stations, or if plants are to be distributed among multiple automation stations. You can also consider other criteria, such as available panel sizes.

Room automation

You can choose solutions with LON and/or KNX. You can choose predefined solutions with drag & drop and then equip them with the required field devices. This way you can create a sample room and replicate it as required.

Third-party devices

You can integrate third-party devices with protocols, such as LON, KNX, ModBus, M-bus or OPC, on all levels.

4.11 Desigo Xworks Plus (XWP)

You can edit project data in the Xworks Plus Editors.

See *Getting Started: Desigo Xworks Plus* (CM110629).

Xworks Project Manager

The Xworks Project Manager lets you:

- Create, open and archive projects
- Check in/out project data for parallel engineering from the Branch Office Server (BOS)
- Define PXC automation stations, control units and Desigo CC. The automation stations are not engineered here, but only used in the documentation and considered during the network check.
- Define rough network overviews (network data) and control cabinet assignments (panel data)
- Define further project data, data and automation stations for RXB and Desigo room automation
- Create control cabinet assignments, that is, group automation stations to control cabinets. This way you can export data and create documentation per control cabinet.
- View locally available projects. There is no connection to the Branch Office Server (BOS) in this mode.
- View the properties of an object, e.g., a network, an automation station, etc.

The screenshot shows the Desigo Xworks Plus Project Manager interface. The main window is titled 'Local Projects' and displays a tree view of project data. The tree view shows a project named 'Project BT' with sub-items: AutomationStationData, ManagementStationData, OtherSysNodesData, NetworkData, PanelData, and MiscFiles. The main area displays a table of project data with columns for Name, Description, and Info. The table lists various project components such as ProjectData, AutomationStationData (AS01, AS02), ManagementStationData (MS01, FOD01), NetworkData (NET01, -SEG01), and PanelData (Plant room 123, -Panel 23-7). The right pane shows the properties of the selected project, including General, Manufacturer Data, Project, and System information. The bottom status bar shows a job progress for 'Archive Archive 1' with a progress of 100% and the message 'Archive Archive 1 has terminated.'

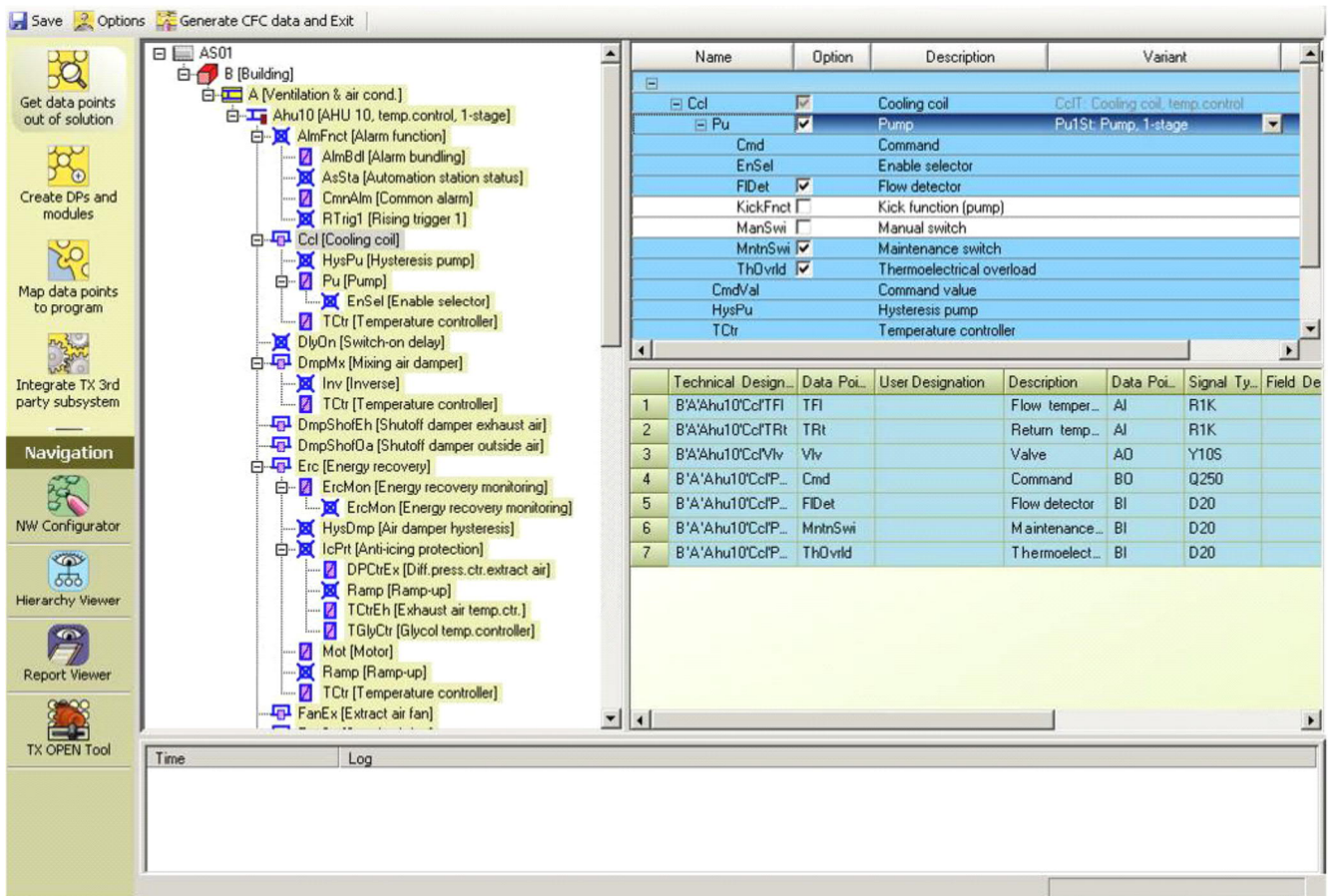
| Name | Description | Info |
|--|--------------------------------|----------------------------|
| <input type="checkbox"/> ProjectData | | |
| <input type="checkbox"/> Project BT | Project 4.0 | Zaehlerweg 3Kamero |
| <input type="checkbox"/> AutomationStationData | | |
| <input type="checkbox"/> AS01 | PXC Contr. 01 | PXC100-E.D + PXA40-W0 / Df |
| <input type="checkbox"/> AS02 | PXC Contr. 02 | PXC200-E.D / DESIGO V4.0 |
| <input type="checkbox"/> ManagementStationData | | |
| <input type="checkbox"/> MS01 | Management Station 01 | |
| <input type="checkbox"/> FOD01 | Field Operator Unit 01 | |
| <input type="checkbox"/> NetworkData | | |
| <input type="checkbox"/> NET01 | BACnet Network 01 | |
| <input type="checkbox"/> -SEG01 | UDPIP Segment 01 | |
| <input type="checkbox"/> PanelData | | |
| <input type="checkbox"/> Plant room 123 | Plant room first floor room 23 | |
| <input type="checkbox"/> -Panel 23-7 | Panel 7 in Plant room 1.23 | |

| General | |
|-------------------|-----------------------|
| Description | Internetwork |
| Name | INET |
| Manufacturer Data | |
| Company | BT |
| Country | Switzerland |
| Region | Zug |
| Project | |
| Addresses | Zaehlerweg 3, BT |
| Archive | |
| Authors | Kamero, PM |
| Description | Project 4.0 |
| Name | Project BT |
| Number | 1123-456 |
| Version | |
| System | |
| Character Set | ISO-8859-1 (ISO Latin |

| Start Time | Duration | Name | Progress | StatusText |
|-------------------|----------|-------------------|----------|-----------------------------------|
| 28.11.08 10:34:29 | 0:00:02 | Archive Archive 1 | 100% | Archive Archive 1 has terminated. |

Xworks Point Configurator

The Xworks Point Configurator lets you define the functions of an automation station. You can insert solutions for the object plant, plant section, aggregate and component into this technical hierarchy. You can configure prebuilt verified solutions using options (leaving out) or variants (options). After you select and configure the solution, the program is automatically created.



The Solution Browser lets you select and configure a plant.

- The tree view shows all selected objects of the plant.
- The configuration view shows all possible options and variants for the selected object.
- The data point window shows all I/Os of the selected object.

You can configure I/Os and I/O modules and connect I/O channels with the I/Os.

You can design the integration of the room automation and the third party integration. The import function lets you integrate third party data points on the automation level. You can import data point information via a standardized interface (SDF format). The BACnet reference browser lets you address BACnet references. You can import BACnet references via a standardized EDE import file (CSV or XLS format).

Xworks Hierarchy Viewer

The Xworks Hierarchy Viewer lets you verify the technical hierarchy of an automation station or entire project. Conflicts in the technical hierarchy are displayed.

The screenshot displays the Xworks Network Configurator interface. On the left, a navigation pane shows options like 'View FX Hierarchy', 'View Technical Hierarchy', 'UD Editor', 'NW Configurator', 'Point Configurator', and 'Define program'. The main area is divided into two panes. The top-left pane shows a technical hierarchy tree for 'Site01 [Site 01]', including 'AS01 [PXC Contr. 01]', 'B [Building]', 'A [Ventilation & air cond.]', 'Global [Global]', 'AS02 [PXC Contr. 02]', and another 'B [Building]' and 'Global [Global]'. The top-right pane displays a table with the following columns: Parent, Short Name, DP Type, AS Name, Description, and UD. The table lists various data points such as 'CDmd', 'DlyOn', 'ElLdEn', 'FlEh', 'FlEx', 'FlOs', 'FlSu', 'FireDet', 'HDmd', 'OpModMan', 'OpModSwi', 'PltCl', 'Sched', 'SmextEh', 'SmextSu', 'SpC', 'SpCorr', 'SpH', 'SpHDmd', 'TEx', 'TOa', 'TSu', 'AlmLmp1', 'AlmLmp2', 'AsSta', and 'CmnAlm'. The bottom pane is an 'Output Window' showing several error messages: 'The technical designation of the Firmware Block "GlobalCln1" of the site "Site01" is not unique!', 'The technical designation of the Firmware Block "GlobalCln2" of the site "Site01" is not unique!', 'The technical designation of the Firmware Block "GlobalCln3" of the site "Site01" is not unique!', 'The technical designation of the Firmware Block "GlobalNotiCl11" of the site "Site01" is not unique!', 'The technical designation of the Firmware Block "GlobalNotiCl12" of the site "Site01" is not unique!', and 'The technical designation of the Firmware Block "GlobalNotiCl13" of the site "Site01" is not unique!'.

| Parent | Short Name | DP Type | AS Name | Description | UD |
|-----------|------------|---------|---------|-----------------------------|----|
| B'A'Ahu10 | CDmd | BVAL_0_ | AS01 | Cooling demand | |
| B'A'Ahu10 | DlyOn | TIMER | AS01 | Switch-on delay | |
| B'A'Ahu10 | ElLdEn | BI | AS01 | Enable electrical load | |
| B'A'Ahu10 | FlEh | BI | AS01 | Exhaust air filter | |
| B'A'Ahu10 | FlEx | BI | AS01 | Extract air filter | |
| B'A'Ahu10 | FlOs | BI | AS01 | Outside air filter | |
| B'A'Ahu10 | FlSu | BI | AS01 | Supply air filter | |
| B'A'Ahu10 | FireDet | BI | AS01 | Fire detector | |
| B'A'Ahu10 | HDmd | BVAL_0_ | AS01 | Heating demand | |
| B'A'Ahu10 | OpModMan | MVAL_ | AS01 | Manual operating mode | |
| B'A'Ahu10 | OpModSwi | MI | AS01 | Operating mode switch | |
| B'A'Ahu10 | PltCl | CMD_C_ | AS01 | Plant control | |
| B'A'Ahu10 | Sched | BSCHEd | AS01 | Scheduler | |
| B'A'Ahu10 | SmextEh | BI | AS01 | Smoke extr. for exhaust air | |
| B'A'Ahu10 | SmextSu | BI | AS01 | Smoke extr. for supply air | |
| B'A'Ahu10 | SpC | AVAL_0_ | AS01 | Setpoint for cooling | |
| B'A'Ahu10 | SpCorr | AVAL_0_ | AS01 | Setpoint correction | |
| B'A'Ahu10 | SpH | AVAL_0_ | AS01 | Setpoint for heating | |
| B'A'Ahu10 | SpHDmd | AVAL_0_ | AS01 | Setpoint heating demand | |
| B'A'Ahu10 | TEx | AI | AS01 | Extract air temperature | |
| B'A'Ahu10 | TOa | AI | AS01 | Outside air temperature | |
| B'A'Ahu10 | TSu | AI | AS01 | Supply air temperature | |
| B'A'Ahu10 | AlmLmp1 | BO | AS01 | Alarm lamp 1 | |
| B'A'Ahu10 | AlmLmp2 | BO | AS01 | Alarm lamp 2 | |
| B'A'Ahu10 | AsSta | AS_STA | AS01 | Automation station status | |
| B'A'Ahu10 | CmnAlm | CMN_A_ | AS01 | Common alarm | |
| B'A'Ahu10 | Over | BVAL_0_ | AS01 | Overload | |

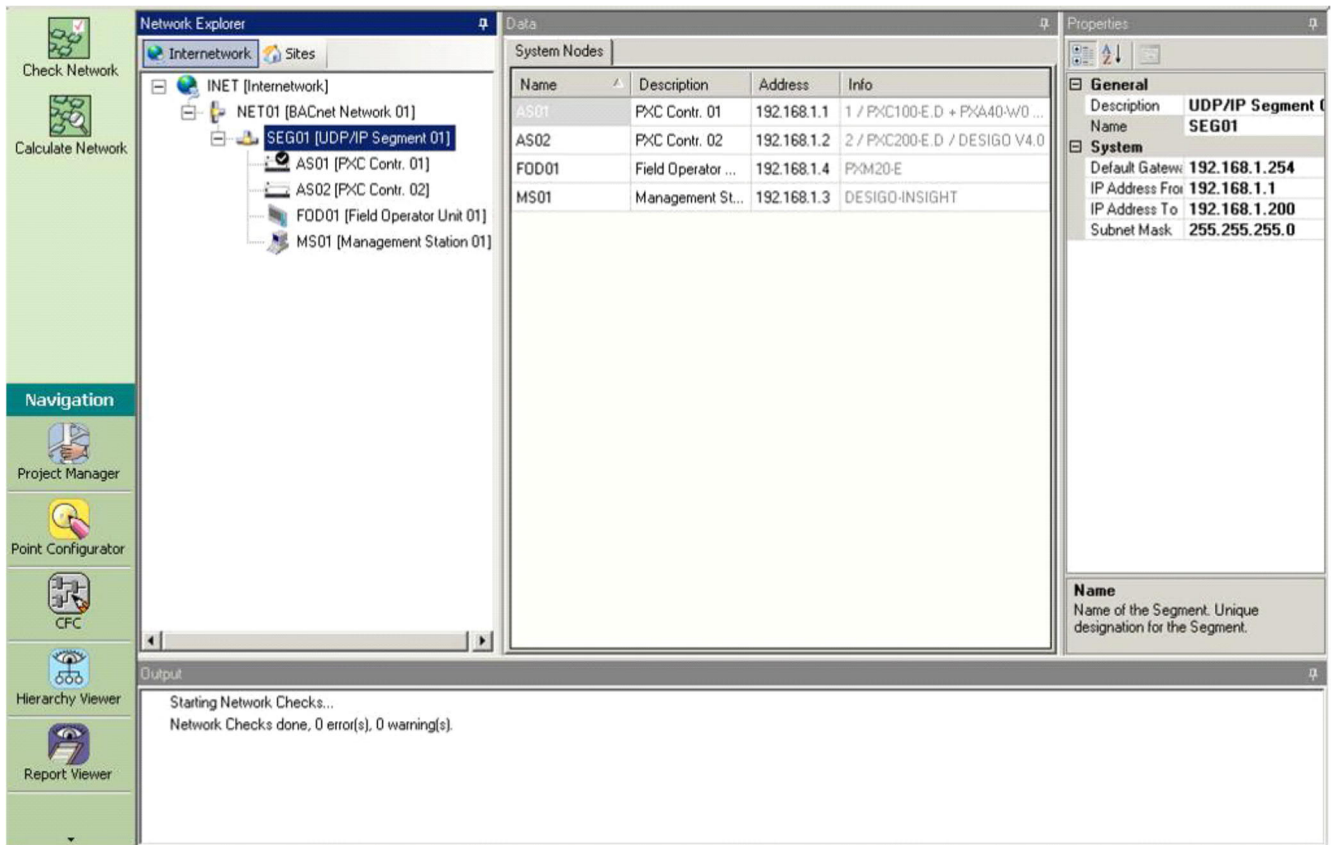
The Xworks Hierarchy Viewer shows the technical hierarchy per PX and the technical hierarchy as it is shown, e.g., in the generic view in Design CC.

You can define the user designation (UD) and the free designation (FD). You can define the structure of the user designation with the field lengths and the separators and assign the data points in the structure of the user designation. You can verify the consistency of the user designation and free designation in the entire project and assign the technical designation (TD = default value), the user designation (UD) or the free designation (FD) to the object name (ON).

Xworks Network Configurator

The Xworks Network Configurator lets you define the network topology. You can define LON, IP networks and network segments, assign and address automation stations to the corresponding segments, and define automation stations and routers.

You can define several sites in a project. The network check verifies all sites in the project. You can verify all automation stations that have been defined in the Automation Building Tool (ABT) for correct and unique addresses, and document it in the network report.



Programming in Xworks Plus

When the technical hierarchy and the automation station are defined, and the I/Os are configured and addressed, you can create a program that corresponds to the selected and configured solutions from the Xworks Point Configurator.

If you use the solution library, you do not have to program in CFC.

Workflow

The workflow for creating programs usually runs as follows.

Workflow in the Xworks Point Configurator

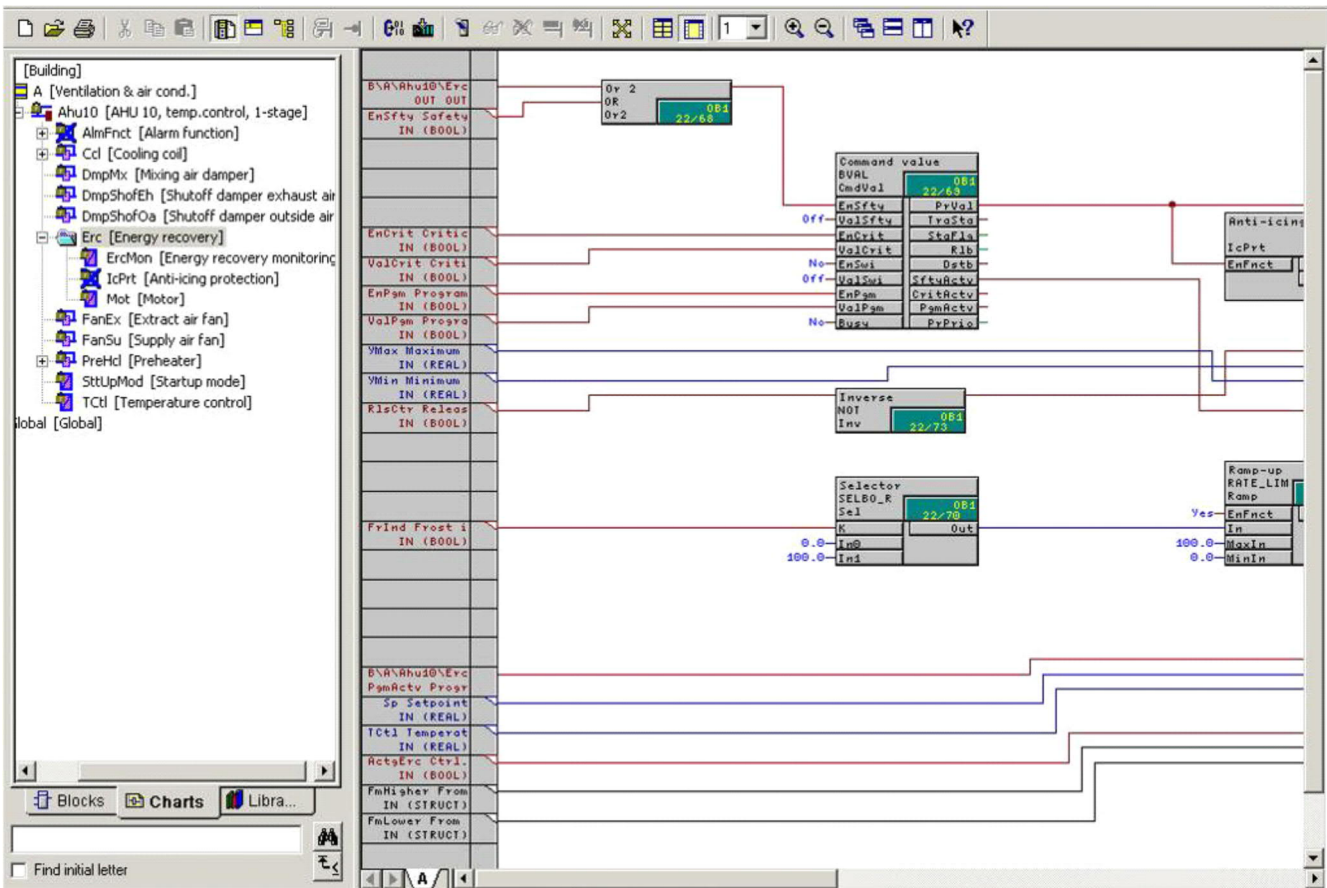
- Select PXC hardware (compact or modular)
- Select and configure solutions
- Configure data points: data point type, signal type and conversion type to field device
- Create and change programs

Workflow in the CFC Classic editor

- Define timer program
- Parameterize alarm behavior and I/Os
- Provide data signals for the energy exchange between different plants
- Transfer plans (create programs)
- load programs
- Carry out commissioning
- Test programs
- Create documentation: data point lists, device plaques, commissioning lists, print parameter lists, etc.

CFC Classic editor

The CFC Classic editor (Continuous Flow Chart) is a graphic tool for creating plans. The CFC Classic editor lets you create and change programs. A CFC plan consists of function blocks and connections.



The CFC Classic editor shows all blocks that are used in the program, nested plans, all available CFC block libraries and the selected plan with the plan interfaces to other plans. This view is available offline for programming and online for checking the signal flow. The CFC Classic editor lets you compile programs, that is, create loadable programs.

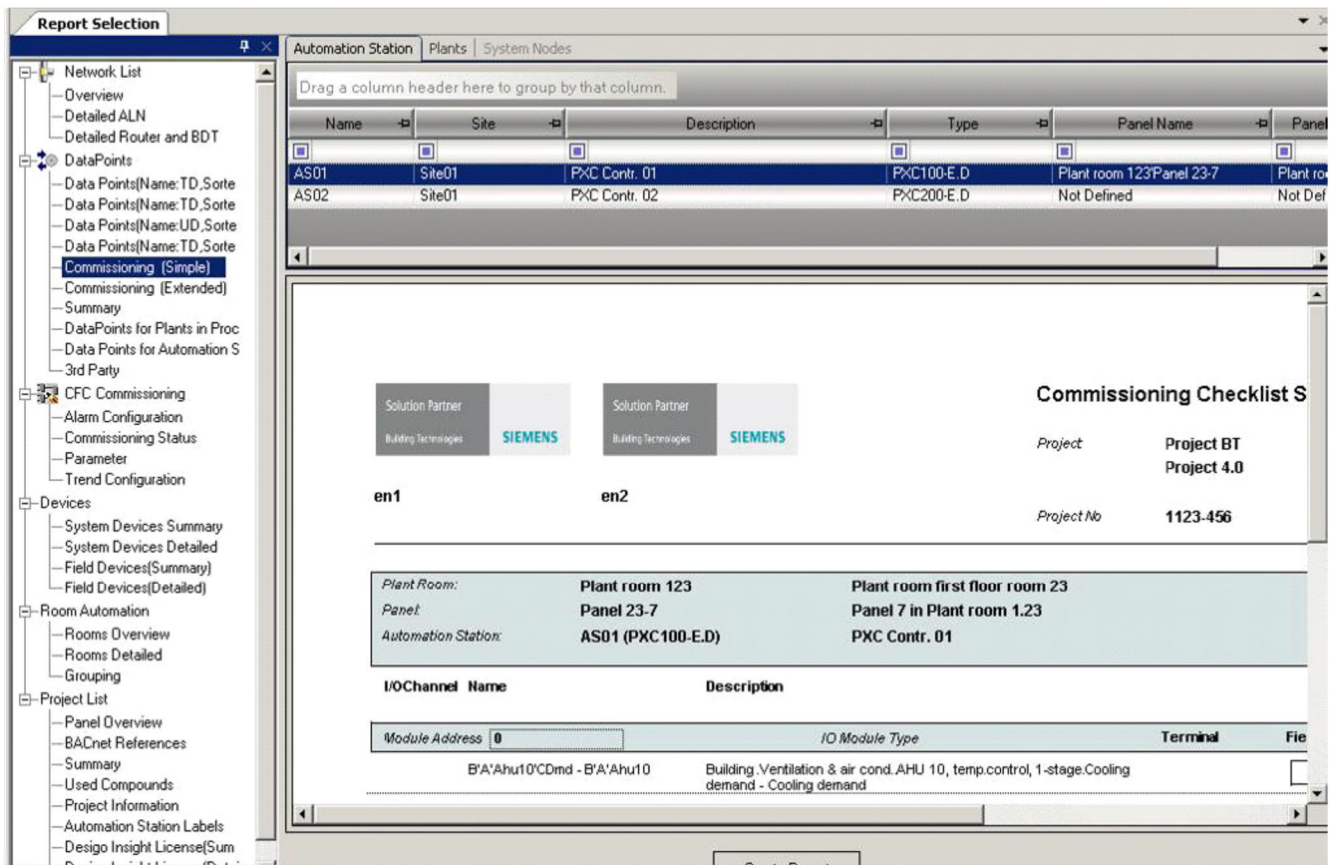
Additional editors

In addition to the graphic programming, you can configure the programs in the following editors:

- Parameter Editor: Lets you parameterize attributes.
- I/O Address Editor: Shows all I/Os of an automation station.
- Plant Control Editor: Lets you configure the plant controls for ventilation and energy generation.
- Solution Configurator: Lets you configure solutions, that are from the CFC library or have been generated from the Xworks Project Configurator.
- Simulation: Lets you simulate programs of a modular automation station without hardware on the computer.
- Alarm display: Continuous update and local caching of all alarm status changes during commissioning. Lets you view, acknowledge and reset alarms.

Xworks Report Manager

The Xworks Report Manager offers comprehensive customer documentation and supports project staff during project handover. The customer can check the documents and after handover the customer is supported during operations, e.g., when handling alarms and interruptions.



You can select one or more automation stations for the documentation, per automation station, plant or system node. You can select document templates and verify reports in a preview.

Designo Point Test (DPT)

Designo Point Test lets you test data points during commissioning of a Designo PX automation station. To carry out a data point test with configured I/O modules, you must download the I/O configuration file for the modular PX automation stations in the empty PX automation station.

BIM Tool

The BIM Tool is used for TX-I/O modules that are integrated with a BIM on automation stations. The BIM was used on old automation stations for integrating I/O modules.

TX Open Tool

The TX Open Tool lets you configure TX Open modules. You can define the TX Open integration and commission the TX Open modules. To commission TX Open, load a configuration into the modules with the TX Open Tool.

See *TX Open Tool online help* (CM111005).

PX KNX Tool (room automation)

See chapter *Room automation*.

HVAC Integrated Tool (HIT)

HIT lets you design HVAC plants. HIT lets you to select and document any HVAC control device as an individual product or in a system configuration. Using its library of over 300 preconfigured HVAC applications for standard controllers (Synco™, Sigmagyr™, and RXB) HIT generates a comprehensive specification including plant diagram, list of material, technical documentation for each device, and pricing.

4.12 Designo Automation Building Tool (ABT)

The Designo Automation Building Tool (ABT) is used for engineering and commissioning Designo room automation.

Project data storage in a Desigo project is handled by Xworks Plus (XWP), that is, you can create a customer project in XWP and check it in to the Branch Office Server (BOS) using Xworks Project Manager. XWP is also used in the Desigo room automation project to carry out the network check and to create the network documentation. Some project reports, which also encompass the Desigo room automation stations are created in XWP.

ABT Site > Projects

In ABT Site > Projects you create projects and define project settings.

ABT Site > Building

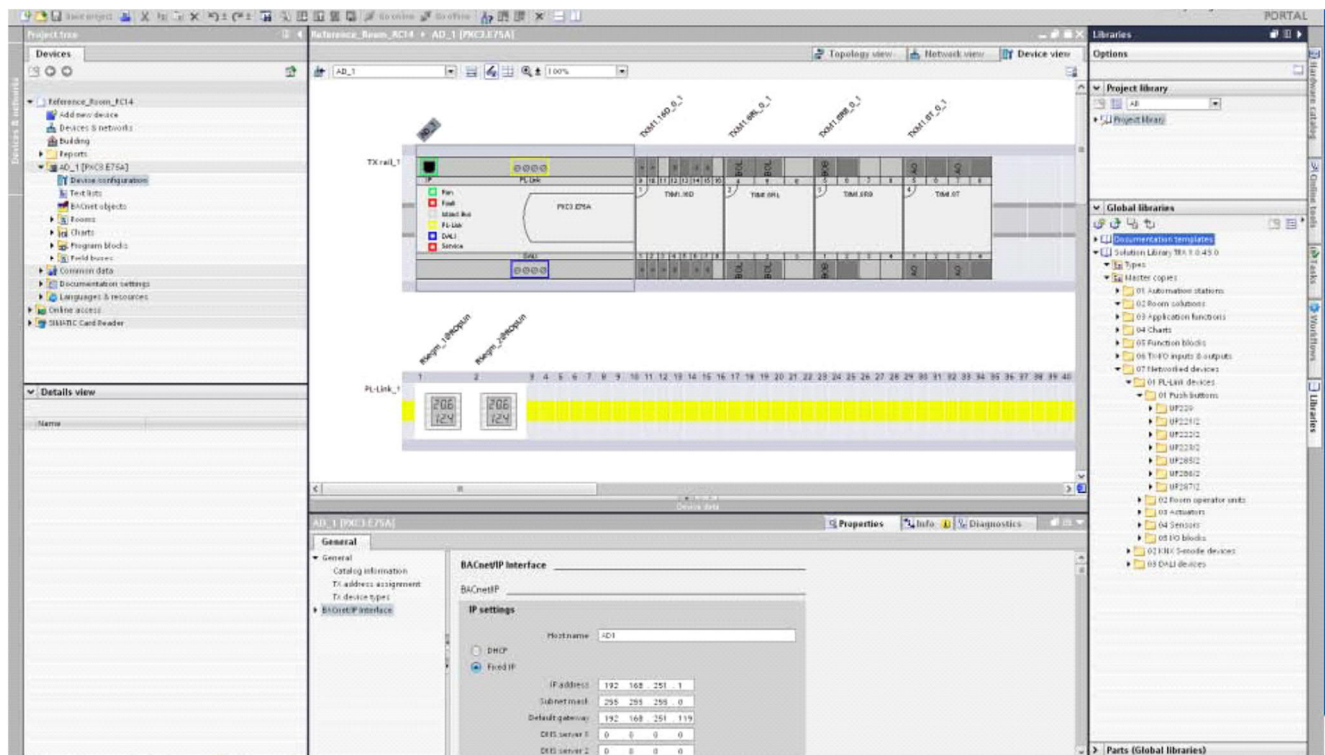
In ABT Site > Building you create building topologies. The topology shows the assignment of room segments and rooms to floors and buildings.

You can define the grouping hierarchies for the central functions and assign the grouping members to the grouping masters. You can create a user designation with a user hierarchy.

ABT Pro

In ABT Pro you program automation stations (project-specific solution). Project-specific solutions can be created in the Center of Competence (CoC) and used as project-specific types in the project in ABT Site.

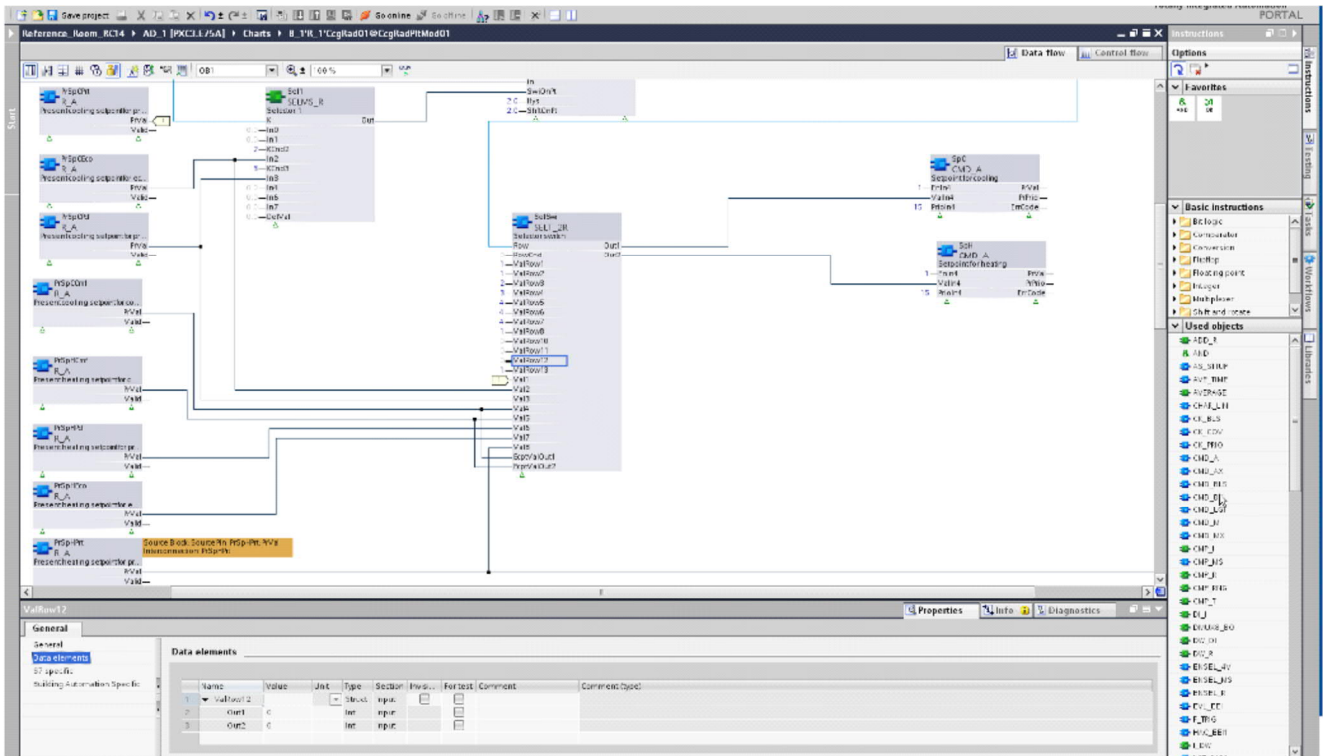
ABT Pro contains the CFC Plus editor for programming in CFC.



ABT Pro shows:

- The automation station objects
- The hardware view of the automation station
- The properties of the selected object
- The project-specific library
- Installed libraries

In the ABT Pro editors you configure room applications, rooms and BACnet objects. In the CFC Plus editor you can program with CFC blocks. The CFC plan contains CFC blocks and connections. A CFC block library is available. ABT Pro is based on the Siemens TIA portal.



ABT Site > Configuration

In ABT Site > Configuration you configure preloaded application types or project-specific types.

ABT Site > Startup

In ABT Site > Startup you scan networks, load configurations and read back parameters.

ABT-SSA

In ABT-SSA (Setup & Service Assistant) you commission I/Os and carry out the point test. See *Designo TRA - Setup & Service Assistant (CM11105)*.

In ABT-SSA you can:

- Assign network points (DALI to device), make points available
- Test if the points work
- Define parameters, e.g., time, desired value, default value, etc.

4.13 Programming in D-MAP

Programming is based on D-MAP principles (Designo Modular Application Programming), where you assemble blocks into compounds and then you build hierarchically structured solutions using those compounds.

- In Xworks Plus (XWP) you program in the CFC Classic editor.
- In the Automation Building Tool (ABT) you program in the CFC Plus editor.

The CFC editors have a different look and feel. Their basic functions and basic library blocks are almost identical.

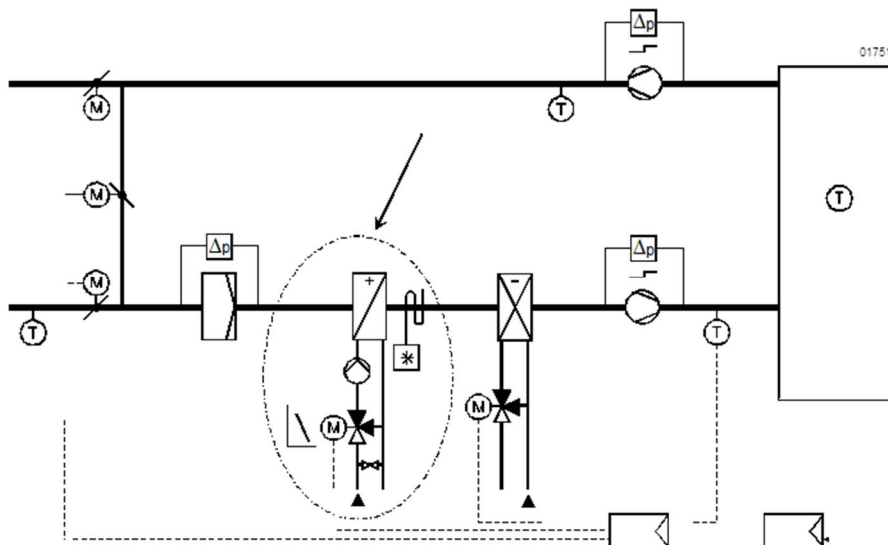
Programming in XWP for Designo PX

The Program View describes the concepts and elements on which D-MAP is based: libraries, compounds, blocks, variables, data types and attributes.

The P&I diagram

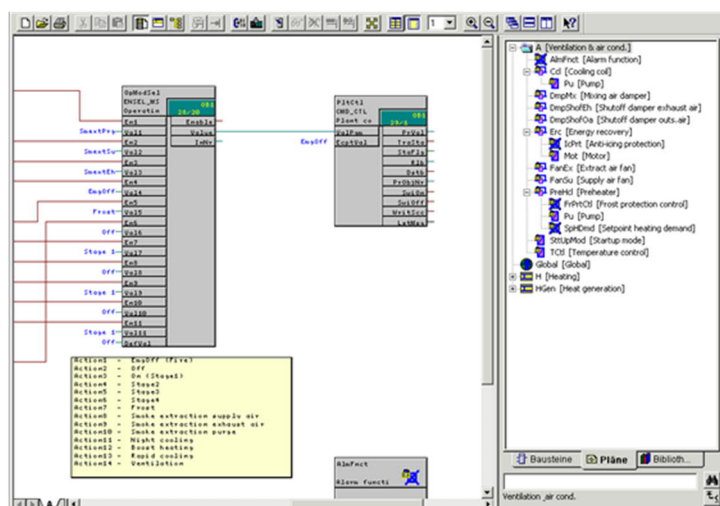
The Program View is based on the P&I (Process & Instrumentation) diagram. The P&I diagram illustrates the plant and the associated instrumentation in the form of a principles diagram.

The following figure shows a simplified P&I diagram of a partial air conditioning system. The heating coil and its components, including the automation station sequence, are encircled.



XWP

XWP is the programming tool for the PX automation station and incorporates all system elements. XWP shows the structural view of the system with the plant, partial plant, aggregates, and components, and, e.g., the compound functional unit for a valve.



Programming in ABT for Design room automation

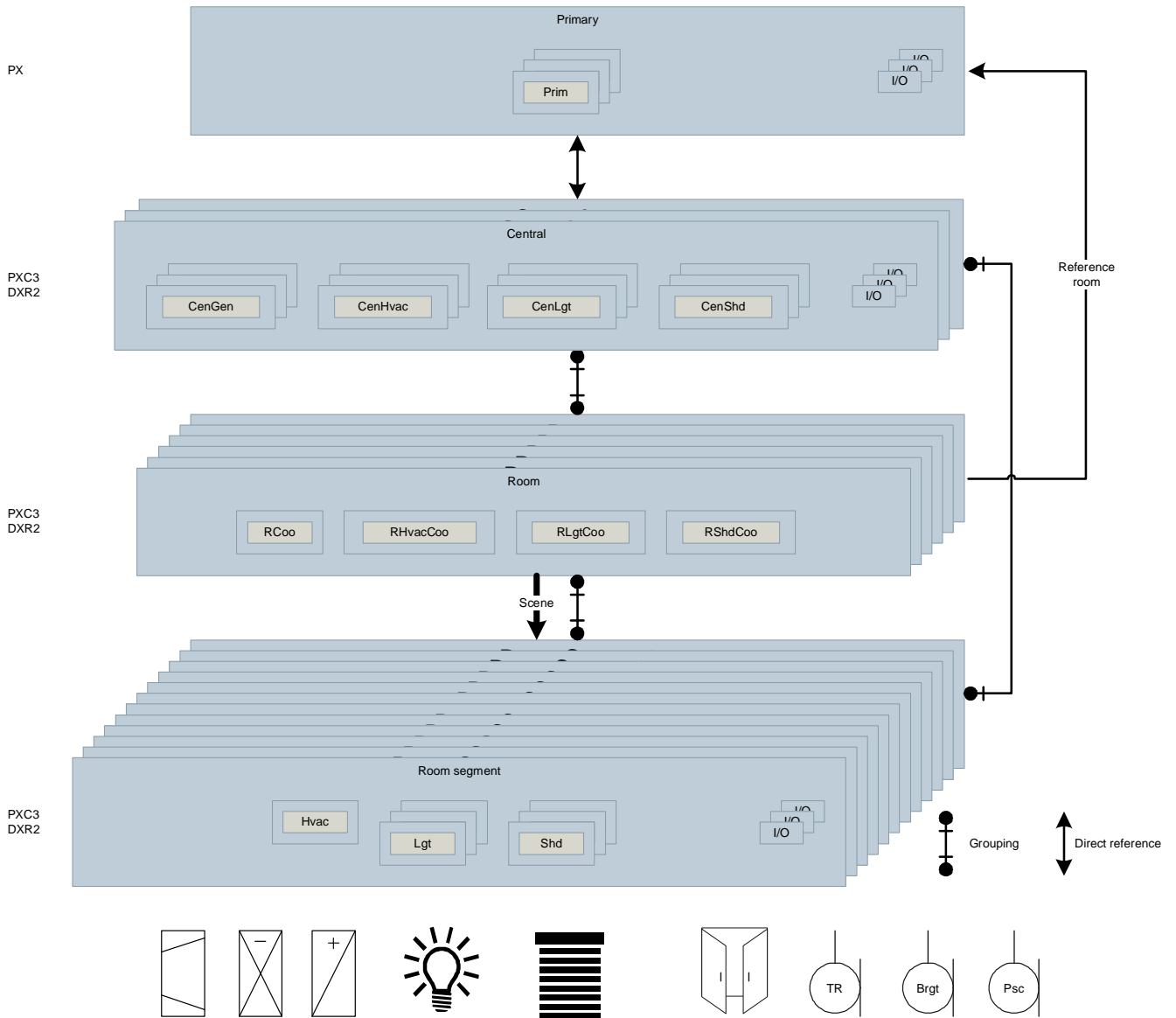
In Design room automation, the application architecture comprises the following elements:

- Hardware configuration: Description of device configurations of the PXC3 automation station
- BACnet description with field device configuration for TX-I/O, KNX PL-Link and DALI
- Automation program: Application description comprising application functions, I/Os and CFC charts

This division lets you define application functions or CFC charts independent of the hardware. The division is also reflected in the loadable units in the tool.

The program view describes the basic concepts and elements for programming for Design room automation: Libraries, CFC charts, blocks, variables, data types, configuration extensions and attributes.

In Design room automation, a program contains the application function (e.g., the lighting function), the associated CFC charts (e.g., the chart for manual control), and the I/O blocks (e.g., the luminaries and buttons).

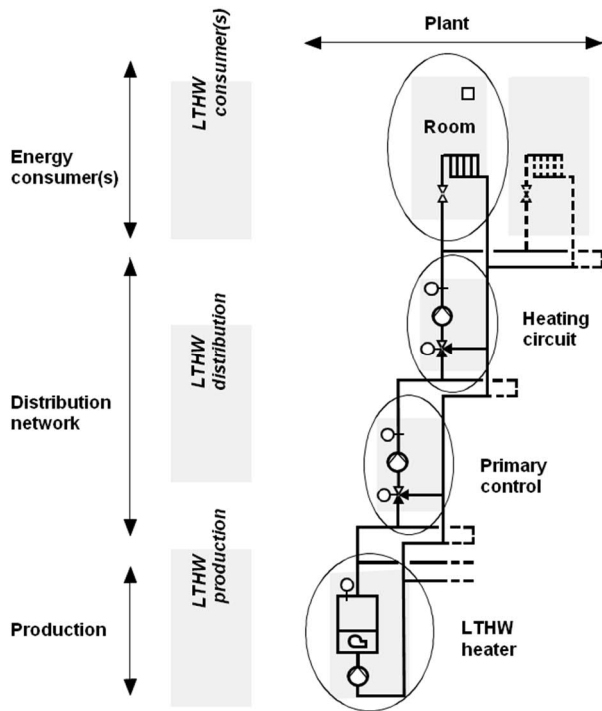


5 Control concept

Supply chain model

In building automation and control, media, such as warm water, cold water, warm air, and cold air are generated using energy, such as oil, gas, and electricity, and distributed to consumers.

Each medium can be assigned a supply chain. The supply chain starts at the generation or handling of the medium. The distribution system then transports the medium to one or several consumers. A supply chain for building services systems comprises the following links:



Consumers

The consumer supplies the energy contained in the hot water medium to the room as per the requested demand (e.g., via a radiator).

Distribution

The distribution system transports the medium from the producer to the consumer and adjusts it to the individual requirements (minimum losses).

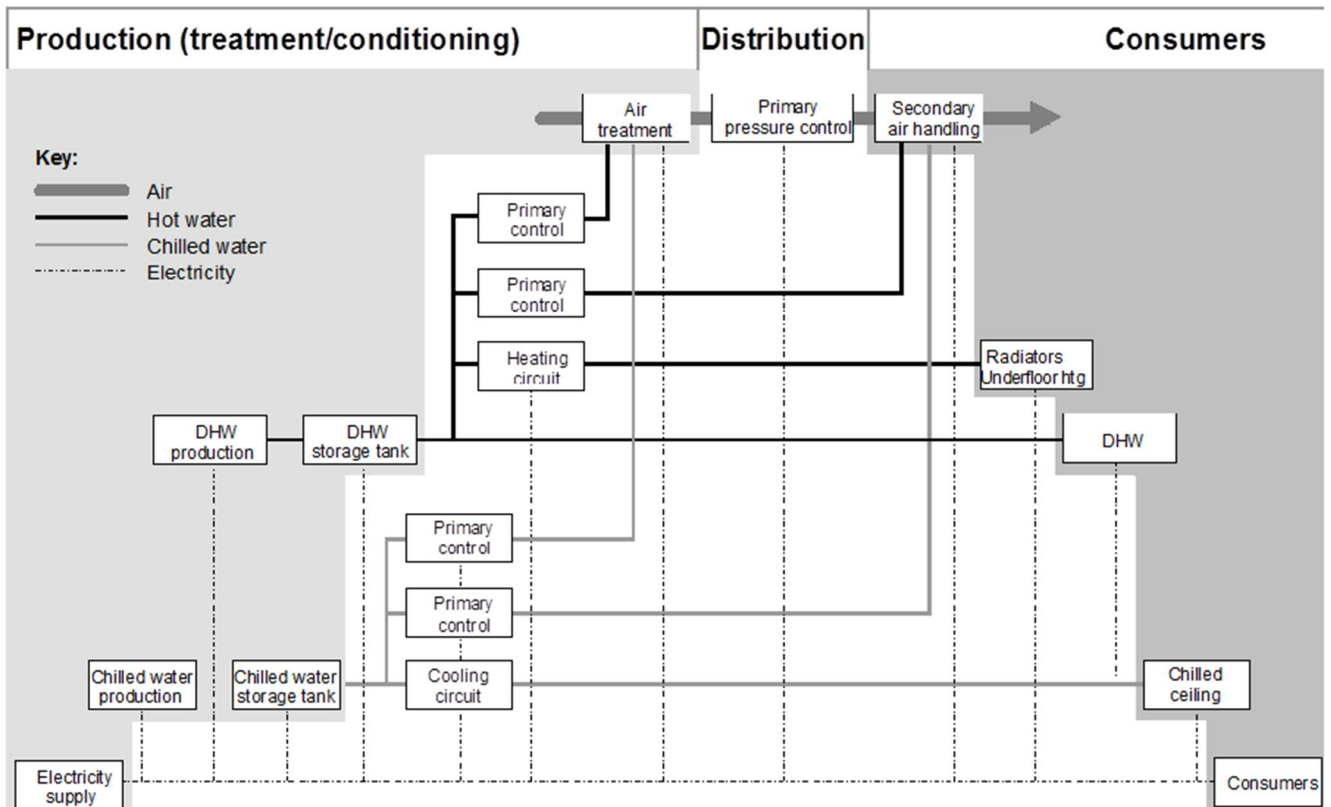
Production

The production consists of a boiler where hot water is treated by means of energy (e.g., heating oil, gas) and provided to the process.

Supply chains of various media

The following illustration shows a schematic view of the supply chains for the media air, hot water, and cold water with their respective production (treatment), distribution (e.g., heating circuit, pre-control), and the consumers.

The supply chain for the medium electricity, which normally begins at supply or at production, if electricity is produced on-site (e.g., cogeneration plant, photovoltaic) is also shown.



A tree structure opens to the right for the individual supply chains. In other words, one or more generators supply multiple primary controllers and each primary controller for its part supplies one or more consumers or other primary controllers.

From the air supply chain point-of-view, air treatment is a part of production (handling). From the hot water and cold water point-of-view, air treatment (or air heater/cooler) belongs to consumption.

The air supply chain comprises the central air treatment plant, optionally supplemented by pressurization control and air posttreatment.

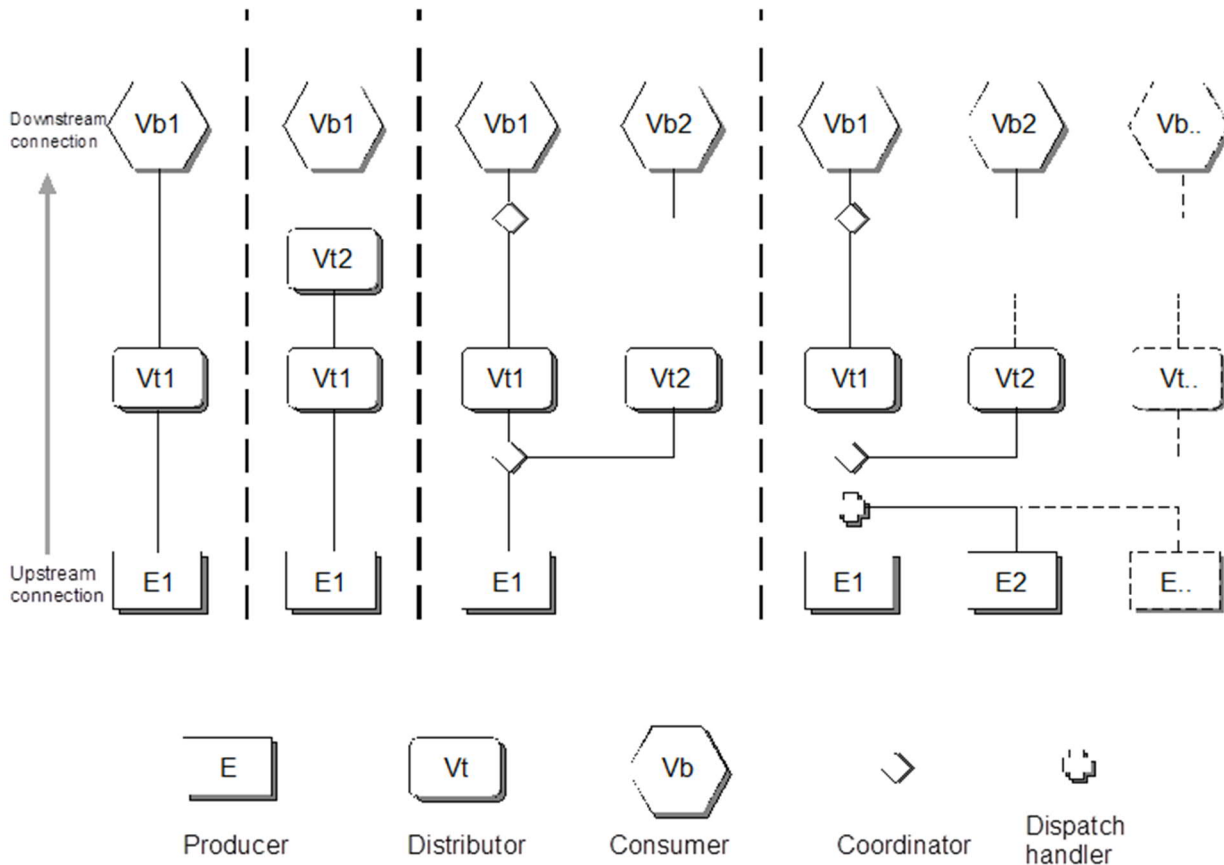
Supply flow

In each supply chain, the medium flows from the producer, through the distribution system to the consumer. This flow within the supply chain is referred to as the supply flow.

Supply chain structure

A supply chain consists of at least one producer and one consumer. It can also have multiple chain links, that is, producers, distributors, and consumers, and be structured as follows:

1. One producer with one distributor and one consumer.
2. One producer with two distributors in series and one consumer.
3. One producer with two distributors in parallel and two consumers in parallel.
4. Multiple producers, distributors, and consumers in parallel.



Producer

In practice, however, there are often multiple producer units, e.g., boilers with the same or similar power, or a mixture of different units, e.g., boiler combined with a solar plant and cogeneration plant (usually with additional storage units).

Logical producer

From the distributor and consumer point-of-view, there is only one single producer within the supply chain, the logical producer, with exactly one supply point as the interface to the distribution network. This logical producer knows nothing about the structure of the distribution network and the connected consumers. Also, neither the distributor nor the consumer knows whether the producer consists of one or multiple units.

Distribution components

The distributor or distribution transports the medium within the supply chain. In this process, energy losses and energy consumption of pumps and fans is to be kept to a minimum.

Conversion

Conversion (transformation) of the medium, e.g., in a heat exchanger, is assigned to a supply chain of distribution. A change of temperature (e.g., pre-control in the heating circuit) is also seen as conversion. Pre-controllers can be arranged in series (cascading).

Consumers

The following consumers, e.g., belong to the various supply chains:

| Supply chain | Consumers |
|--------------|--|
| Hot water | Air treatment and air posttreatment (heating register) Radiators (radiator, convector) Floor heating, domestic hot water heating |
| Cold water | Air treatment and air posttreatment (cooling register) Cooling surface (chilled ceiling) |
| Air | Air posttreatment (dampers) |
| Electricity | HVAC consumers, other consumers |

Coordinator and dispatcher

In addition to the three chain links producer, distributor, and consumer, there are the logical links named coordinator and dispatcher.

Supply chains for a room

You can define different consumer needs for a room, such as heat, refrigeration and fresh air.

Heat demand

The hot water supply chain exists for heat demand. The medium hot water is prepared in hot water generation and distributed via a heating circuit. The heat is emitted to the room as needed via a heating surface. If air is the carrier of heat, this is done via pre-control and air posttreatment.

Refrigeration demand

The cold water supply chain exists for refrigeration demand. The medium cold water is prepared in cold water generation and distributed via a cooling circuit. The refrigeration is emitted to the room as needed via a cooling surface. If air is the carrier of refrigeration, this is done via pre-control and air posttreatment.

Fresh air demand

The need for fresh air is met by the air supply chain, where the medium is produced by the air treatment plant, distributed via the ducting, possibly adjusted to differing requirements of the room by an air posttreatment plant, and transferred to the room via air outlets.

HVAC application architecture

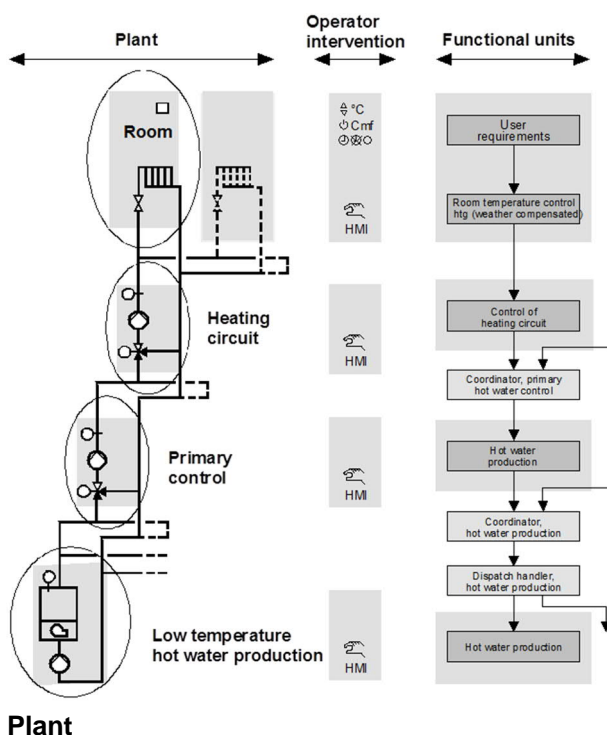
The HVAC application architecture contains an overall view of typical heating, ventilation and air conditioning plants with distributed applications and is based very strongly on the supply chains (energy and substance flows) in building services systems.

- The mutually standardized exchange and re-use of HVAC-relevant demand and coordination signals is possible in distributed applications.
- The HVAC application architecture structures the HVAC functions into meaningful units, interfaces and functional mechanisms.
- The HVAC application architecture is scalable and independent of product and communication standards.

HVAC system view

The consideration and definition of the HVAC application architecture and its functionality gives rise to the HVAC system view, which comprises:

- Plant (primarily HVAC plants)
- Operator interventions
- Functional units



A plant consists of partial plants, aggregates, and components, which, as a rule, form a supply chain with the chain links producer (here: boiler), distributor (pre-control, heating circuit), and consumer (radiator).

Operator interventions

Commands are executed at each link of the chain through operating interventions via HMI commands. The impact on the plant (or the process) takes place via the corresponding function unit and automation station.

Functional units

Functional units represent the software map of chain links and plant elements. The functional units contain all control, monitoring, and limiting functions that are necessary for operation.

Information signals

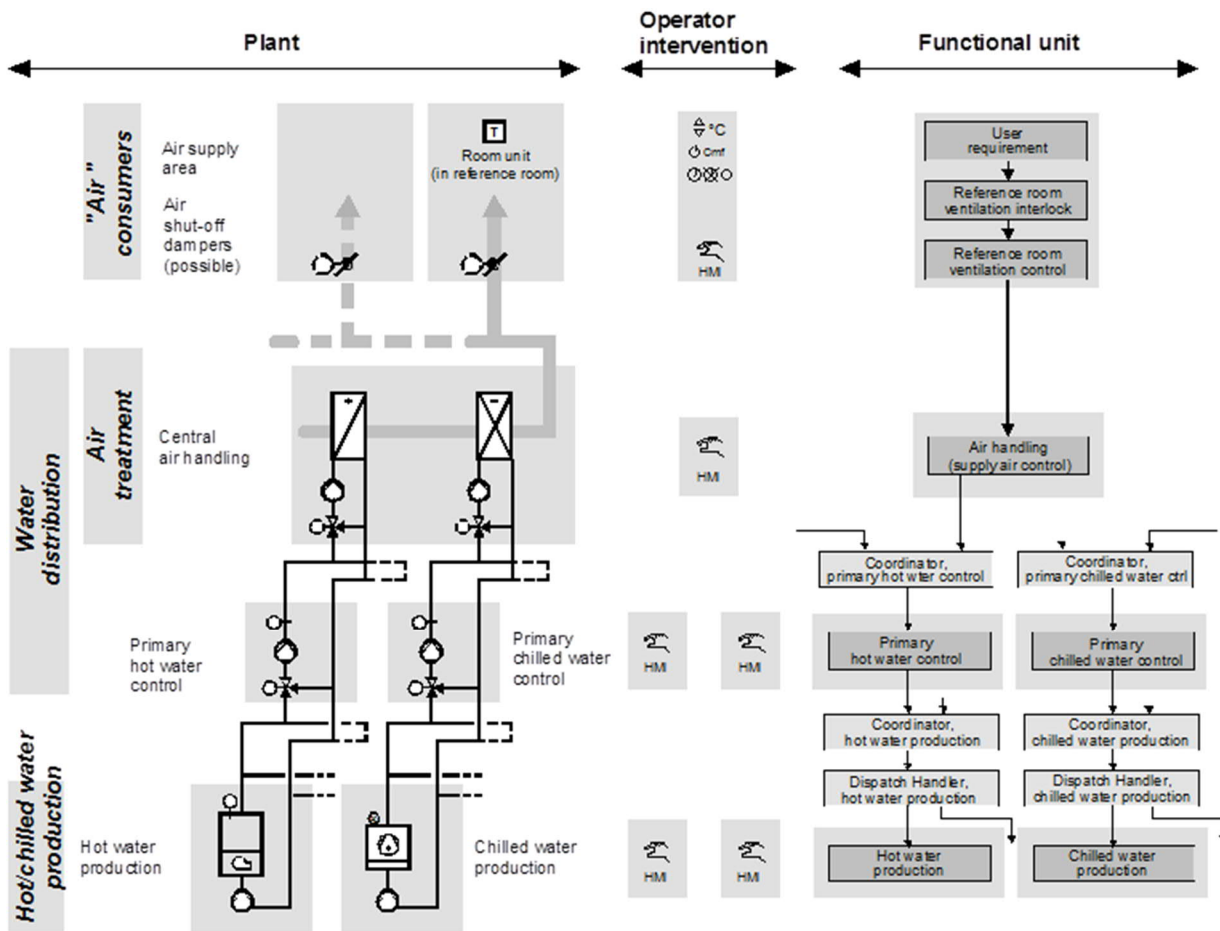
Energy demand information can be passed on implicitly via the medium within the supply chain, e.g., if the hot water supply temperature falls because of a rise in heat consumption, more heat energy must be produced.

Information can also be represented by an explicit signal and transferred via a signal path (e.g., via a bus). The following explicit signals have been defined in the Desigo system:

| Explicit signals | Signal flow | Application |
|------------------|----------------------|---|
| Demand signal | Consumer to producer | A plant functional unit communicates its demand (that is, operating mode, set points) to another partial plant functional unit in the direction of the producer. The demand signal eventually arrives at the producer. |
| Operating signal | Producer to consumer | A plant informs the downstream plants about its currently effective operating state. This signal is only used as an exception and is therefore switched depending on the situation. |
| Override signal | Producer to consumer | The producer demands a certain operating mode from a consumer. Forced signals are more the exception than the rule and are therefore not implemented in sample plants. Forced signals are used for solar plants and wood furnaces among others, where the minimum heat production cannot be controlled. |

In addition to the functional units, there are two further elements that belong to the supply chain on the software side:

- **Coordinator:** The coordinator combines the demand signals of downstream (to supply flow direction) plants and delivers a resultant demand signal to the upstream plants. The coordinator also signalizes the operating state of the upstream plants to the downstream plants.
- **Dispatcher:** The dispatcher determines the demand signals for the producers on the basis of the resultant consumer demand signals. It decides which and how many producers must be activated.



5.1 Control concept and control blocks

The Designo control concept is a set of rules that determine in general terms the principles governing all control, reporting and monitoring operations and the switching interventions in the Designo system. The rule applies to block-internal control (priority array) and to functional interactions among participating blocks.

This specifically deals with:

- Structure and design of control as function blocks
- The hierarchical assignment of the function blocks among themselves
- The function hierarchy within the control chain for the function blocks
- Processing operational and fault messages
- Interventions in monitoring functions
- Impact of emergency switching

The open loop control strategy is based on the exchange of predefined signals between functional units. Each functional unit is an image, or memory map, of an actual element of the plant, e.g., ventilation or boiler plant.

Control functions

The open-loop control functions required for a given element are locally an integral part of the functional unit (e.g., the increase, after a time delay, in the speed of a multi-stage fan, or the demand-based switch-on of a boiler). In each functional unit, various possible requirements are prioritized and evaluated. The resulting operating mode is then passed on to the elements or subordinate functional units. The functional unit already incorporates the I/Os needed for the physical data points.

Structure control functions

In this way, complex control and monitoring functions of a plant can be logically subdivided to allow for clear assignment of the function unit or the real element of the plant. The higher-level control concentrates on the control and monitoring of the overall plant, while the sub-control function units assume internal control and monitoring of the given elements for the function unit.

Standardization of control functions

Moreover, plant security and available was increased through standardized control and monitoring functions which would result in considerable expense using conventional methods.

Standardized control and monitoring functions:

- Unambiguous selection of operating mode
- Uniform fault-related shutdown
- Comprehensive status monitoring
- Switching sequence for ventilation systems
- Output stage control for heat generating plant
- Reporting of local intervention
- Avoidance of unnecessary attempts at switching
- Prevention of inadmissible switching operations
- Protection of plant by preventing switch-on or switch-off

Blocks bound by the control concept

| Function | Block name | Task in the control concept |
|---|----------------------|--|
| Prioritization of influencing variables | ENSEL_BO ENSEL_MS | Collect information for the selection of the resulting plant operating mode. All superposed information are processed by priority resulting in the plant being turned on or off, e.g., smoke extraction switch, frost protection, scheduler program. The blocks are primarily used on the hierarchy level plant/partial plant, but may also be reasonably used, e.g., in aggregates. |
| Command control | CMD_CTL | Superposed control block for sequence control. The block ensures that individual plant aggregates are switched on or off sequentially in a certain order. The block monitors aggregates and can send alarms. It is optimized for controlling air handling plants, but can be used for other applications. The block is used on the hierarchy level plant/partial plant. |
| Power control | PWR_CTL | Superposed control block for power control. The block is used for control and monitoring of the performance of a number of energy producers (multiple boiler systems, refrigeration machines). Depending on the request power demand, energy produces are switch on or off in stages. PWR_CTL is optimized for controlling heating and refrigeration plants. The block is used on the hierarchy level plant/partial plant. |
| I/O blocks with control functionality | BO MO AO | Output blocks implemented per BACnet standard and therefore including a priority mechanism (priority array) that is well suited for control tasks. The priority array [PrioArr] be used through data flow interconnections and BACnet commanding. Moreover, the block integrate the following control functionality: - Motor control (pump, burner, etc.), one- to four-speed [BO, MO] - Fan control, one- to four-speed [BO, MO] |
| Value blocks with control functionality | BVAL MVAL AVAL | Value objects or value blocks are implemented per BACnet standard and therefore includes output blocks via the priority mechanism. These blocks are referred to as data points that can communicate within the system with the I/O modules via BACnet. These blocks are primarily used as the communication interface between superposed control [CMD_CTL, PWR_CTL] and the aggregates. |
| Rotation block | ROT_8 | The Rotation block switches the operating mode on and off for a maximum of eight functional units in accordance with a selected mode of rotation (sequence or hours run). The change of operating mode is based on demand, hours run, occurrence of a fault or manual intervention (override). The block is used to process the functional units (e.g., aggregates or components) as a function of run-time or faults. These blocks are used, e.g., for double pumps, that are changed over based on runtime. |

Control hierarchy

Control hierarchy is the map of the functional assignment and linking of those function blocks included in the control concept for a plant. The structure of the control hierarchy is subject to certain rules. A distinction is drawn between higher-level plant control and local control of the functional units.

Superposed control

Within the hierarchical structure, higher-level control functions are typically assigned to the partial plant level. All the variables which are influencing factors on the overall plant are weighted and combined to give the effecting plant operating mode. In respect of each of the possible plant operating modes, a control strategy can be defined for each underlying functional element. This makes it possible to develop specific plant scenarios, such as fire control, smoke extraction, frost control, on/off-switch control.

Local control

Within the hierarchical structure, local control of the function elements is typically assigned to the partial plant level. The main function of local control is to respond to faults. The functional unit itself determines how the outputs are to be controlled in the event of a fault. Interlocks between functional units (e.g., damper/fan) must be implemented locally. Local control prevents the risk of damage to plant, in the event that the command control parameters are set incorrectly.

The control hierarchy in the following figure considers only the example application for ventilation.

Higher level plant control
 Hierarchy level: Partial plant
 - Plant enable
 - Scheduler
 - Plant switch...

- Response to faults

- Sequence control

- Switch on

- Switch off

- Failsafe function

- Smoke

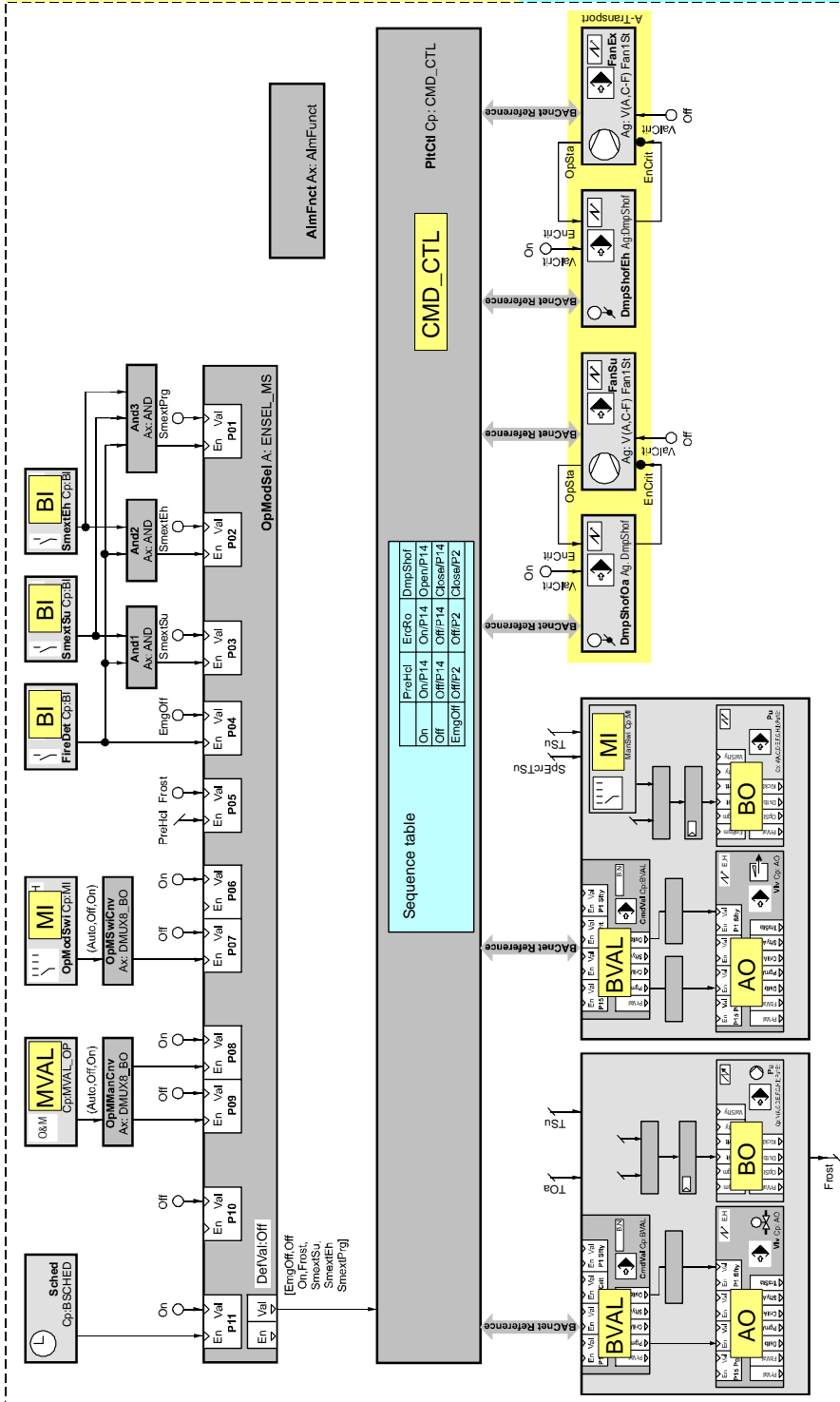
- Frost

- Night cooling

- Purge

Local control
 Hierarchy level: Aggregate
 - Local response to faults
 - Interlocks
 - Failsafe values
 - Local operat. intervention

103983707en



I/O block functions and interfaces

The I/O blocks are the most important blocks in the Designo system. In addition to controlling the hardware, they are responsible for numerous control and monitoring functions. They enable otherwise complex functions to be implemented with just a small number of blocks.

Main functions and interfaces of I/O blocks

| Function | Inputs | Description | AI | AVAL | AO | BI | BVAL | BO | MI | MVAL | MO |
|---|-----------|------------------------|----|------|----|----|------|----|----|------|----|
| Stop transmission of input signal | OoSrv | Out of service | • | • | • | • | • | • | • | • | • |
| Priority mechanism | DefVal | Default value | | • | • | | • | • | | • | • |
| | PrioArr | Priority array | | • | • | | • | • | | • | • |
| Local override | Ovrr | Override | | • | • | | • | • | | • | • |
| | OvrrVal | Override value | | • | • | | • | • | | • | • |
| Alarm value monitoring - Limits - Reference values - Monitoring periods | EnAlm | Alarm enable | • | • | • | • | • | • | • | • | • |
| | HiLm | Upper limit | • | • | • | | | | | | |
| | LoLm | Lower limit | • | • | • | | | | | | |
| | Nz | Neutral zone | • | • | • | | | | | | |
| | RefVal(s) | Reference value | • | | | • | • | | • | • | |
| | TiMonOn | Monit. time switch-on | | | | • | • | • | • | • | |
| | TiMonOff | Monit. time switch-off | | | | • | • | • | • | • | |
| | TiMonDvn | Monit. time deviation | • | • | • | • | • | • | • | • | • |
| Switching delays | DlyOn | Switch-on delay | | | | | • | • | | • | • |
| | DlyOff | Switch-off delay | | | | | • | • | | • | • |
| | TbTiDly | Time delay table | | | | | • | | | • | • |
| Switching action Normal (motor) Release command Trigger Switch Switch with delay | SwiKind | Switch kind | | • | • | | • | • | | • | • |

| Function | Outputs | Description | AI | AVAL | AO | BI | BVAL | BO | MI | MVAL | MO |
|------------------------|----------|------------------------|----|------|----|----|------|----|----|------|----|
| Feedback monitoring | PrVal | Present value | • | • | • | • | • | • | • | • | • |
| | FbVal | Feedback value | | | • | | | • | | | • |
| Reliability monitoring | Rlb | Reliability | • | • | • | • | • | • | • | • | • |
| Fault monitoring | Dstb | Fault | • | • | • | • | • | • | • | • | • |
| Status monitoring | TraSta | Transitional state | | • | • | | • | • | | • | • |
| Priority monitoring | SftyActv | Safety priority Active | | • | • | | • | • | | • | • |
| | CritActv | Critical active | | • | • | | • | • | | • | • |
| | PgmActv | Program active | | • | • | | • | • | | • | • |
| | PrPrio | Present priority | | • | • | | • | • | | • | • |

| Function | Inputs | Description | AI_RED | AO_RED | BI_RED | BO_RED | MI_RED | MO_RED |
|-----------------------------------|---------|----------------|--------|--------|--------|--------|--------|--------|
| Stop transmission of input signal | OoSrv | Out of service | | | | | | |
| | DefVal | Default value | | • | | • | | • |
| Priority mechanism | PrioArr | Priority array | | • | | • | | • |

| Function | Inputs | Description | AI_RED | AO_RED | BI_RED | BO_RED | MI_RED | MO_RED |
|--|-----------|------------------------|--------|--------|--------|--------|--------|--------|
| Local override | Ovrr | Override | | • | | • | | • |
| | OvrrVal | Override value | | • | | • | | • |
| Alarm value monitoring - Limits - Reference values - Monitoring periods | EnAlm | Alarm enable | | | | | | |
| | HiLm | Upper limit | | | | | | |
| | LoLm | Lower limit | | | | | | |
| | Nz | Neutral zone | | | | | | |
| | RefVal(s) | Reference value | | | | | | |
| | TiMonOn | Monit. time switch-on | | | | | | |
| | TiMonOff | Monit. time switch-off | | | | | | |
| Switching delays | DlyOn | Switch-on delay | | | | | | |
| | DlyOff | Switch-off delay | | | | | | |
| | TbTiDly | Table for time delay | | | | | | |
| Switching action - Normal - Release command - Trigger | SwiKind | Switch kind | | • | | • | | • |

| Function | Outputs | Description | AI_RED | AO_RED | BI_RED | BO_RED | MI_RED | MO_RED |
|------------------------|----------|------------------------|--------|--------|--------|--------|--------|--------|
| Feedback monitoring | PrVal | Present value | • | • | • | • | • | • |
| | FbVal | Feedback value | | | | | | |
| Reliability monitoring | Rlb | Reliability | • | • | • | • | • | • |
| Fault monitoring | Dstb | Fault | • | • | • | • | • | • |
| Status monitoring | TraSta | Transitional state | | • | | • | | • |
| Priority monitoring | SftyActv | Safety priority Active | | • | | • | | • |
| | CritActv | Critical active | | • | | • | | • |
| | PgmActv | Program active | | • | | • | | • |
| | PrPrio | Present priority | | • | | • | | • |

Priority mechanism

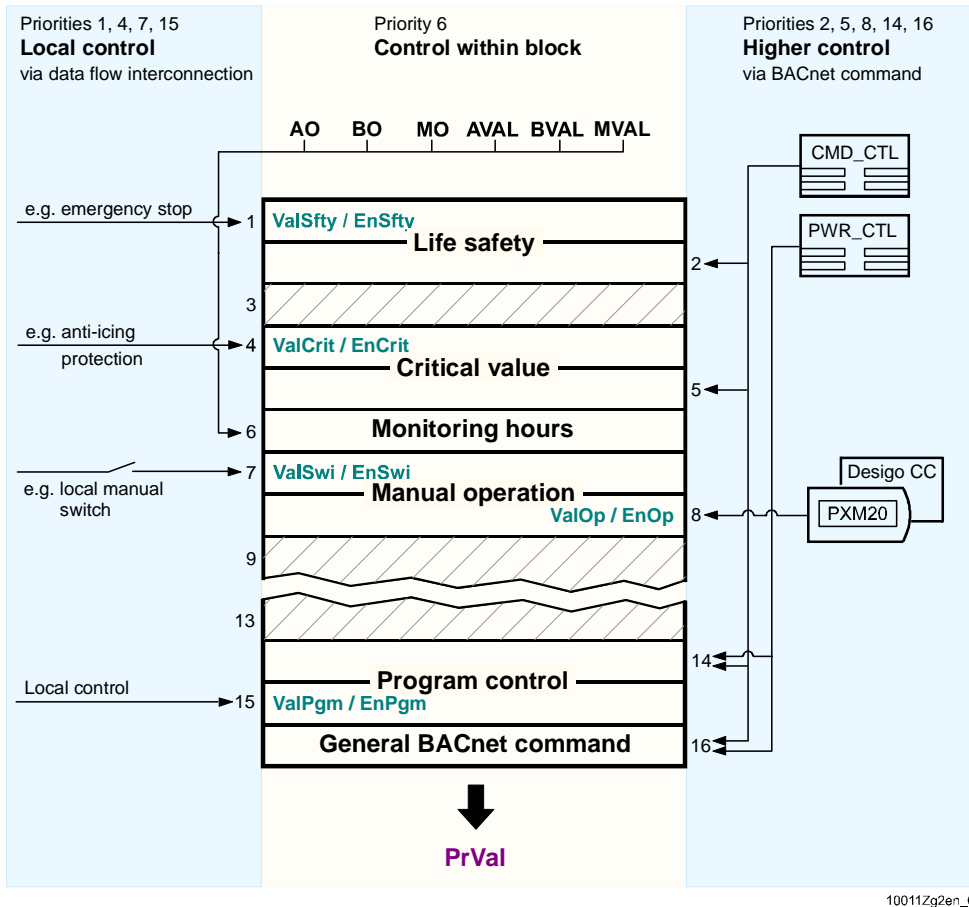
Within the Desigo PX system, the BACnet priority mechanism is used for the I/O output blocks and in the value blocks. This priority mechanism provides a series of prioritized levels at which intervention is possible, for use with the control functions in HVAC plant and the associated components.

The following priority levels are available with blocks AO, BO, MO (and blocks AO_RED, BO_RED, MO_RED) and AVAL, BVAL and MVAL.

| Level | Application | Description |
|----------------|--------------------------------|---|
| Safety level | Life safety | The safety level is assigned the highest priority and is used for the protection of people and equipment. This is where local safety switches and emergency OFF buttons are wired or superimposed commanded, e.g., smoke extraction control or frost control. |
| | Plant safety | |
| Operator level | Local manual intervention | The operator level is where components are overridden manually. Here the operator unit may be used to force the output of an I/O function block. This operation overrides all operations at a lower priority level. |
| | Superposed manual intervention | |

| Level | Application | Description |
|-----------------|---------------------------|---|
| Automatic level | Local control | The automatic level is used for local control functions and for superposed BACnet commanding. |
| | General BACnet commanding | |

The following figure illustrates the structure of [PrioArr] and the influence of local and higher-level control.



10011Zg2en_01

Local override

The override switch overrides the block's switching value and determines in this way the switching value for the field device. Local override has priority over an active manual operation at the same time, that is, priority over local override.

Status monitoring

[AO, BO, MO, AVAL, BVAL, MVAL]

The process is monitored via the feedback signal, and in the case of switching blocks, also via the ramp-up and ramp-down parameters set in [TbTiDly]. If the feedback value deviates from the present value [PrVal] and the delay in [TbTiDly] has not yet expired, the process is in a transitional state. The status monitoring function shows the status at the transient state [TraSta] output. This output can be used to switch on any subsequent components.

Feedback monitoring

[BO, MO]

Monitoring feedback may be based on a data point or a purely internal to the block based on the feedback time parameter.

- Feedback data point available [FbAddr:] = Address
Monitoring is based on the feedback signals. The delays can be defined with the time parameters for switch-on [TiMonOn], switch-off [TiMonOff] and open-circuit [TiMonDvn]. If the feedback signal [FbVal] deviates from the output value [PrVal], an OFFNORMAL alarm will be triggered (provided the alarm function is switched on).
- No feedback data point available [FbAddr:] = empty
Based on the feedback time parameter [TiMonOn/TiMonOff], the output [FbVal] is delayed by [PrVal]. The output [TraSta] signals transition state.

Alarm value monitoring

[AI, AO, AVAL, BI, BVAL, MI, MVAL]

Alarm monitoring is optional and can be enabled using [EnAlm]. Analog limit or switching values can be monitored depending on the block type. The tolerance time [TiMonDvn] to trigger a process alarm can be set. Deviations for switch on and off procedures can be distinguished for switching blocks.

Alarm monitoring can be enabled based on the process or time. You can switch off frost protection for monitoring in summer e.g..

Reliability monitoring

[AI, AI_RED, AO, AO_RED, AVAL, BI, BI_RED, BO, BO_RED, BVAL, MI, MI_RED, MO, MO_RED, MVAL]

The blocks monitor the reliability of input and output sources and configuration errors. A system alarm is generated, e.g., when a source no longer communicates and the cause is displayed on output [Rib]. The disturbance output [Dstb] changes to yes. This output, e.g., can return to the block for the local disturbance to achieve a more secure position using a higher priority. Reliability monitoring can be switched off using [OoServ], which may make sense for defective or faulty hardware.

Reliability monitoring is always active for the RED blocks since no [OaServ] is available. Superposed control does not distinguish this state and plant safety is not provided under certain circumstances.

Minimum switching times

[BO, BVAL, MO, MVAL]

The minimum time on [TiOnMin] and the minimum time off [TiOffMin] may be defined to reduce switching frequency. For a switch on or off command, is written to [PrioArr] as priority 6 and maintained there during the defined switching period. No lower priority can change the switching value during this time frame.

Switch-on and switch-off delay

[BO, BVAL, MO, MVAL]

To delay switch on or off for elements [DlyOn/DlyOff], e.g., to implement a pump run-on to extract residual heat. For a switch on or off command, the corresponding switching value is written to [PrioArr] as priority 6 and maintained there during the defined switching period. No lower priority can change the switching value during this time frame.

Ramp-up/down time

Runtimes for ramp-up and down

[BO, BVAL, MO, MVAL]

The runtime of a damper or the coasting time for a multi-speed motor can be defined in table [TbTiDly] to display or evaluate a transient state [TraSta]. The time parameter can also influence the switching response depending on the switch kind [SwiKind] used.

Plant fault

The block independently recognizes faults and reports them to the defined alarm class [AlmCl], which for its part is responsible for distributing the alarms to alarm receivers. Depending on the alarm function [AlmFnct] set in the block, you may have to acknowledge the alarm and reset it after eliminating the alarm.

The faulted block on output [Dstb] is reset only after the user action is run. The plant ramps up only after an alarm reset since both the local control, as a rule, with this output is blocked for a fault and as superposed control triggered a plant fault.

The alarm reset can be triggered:

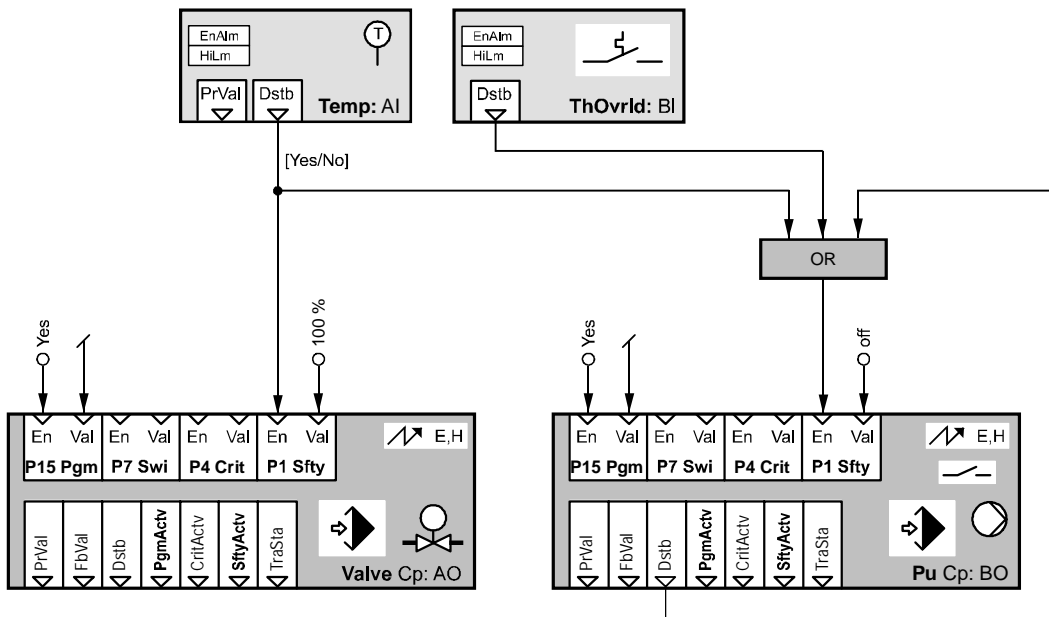
- By triggering a common reset in the common alarm block CMN_ALM
- Via an alarm client

5.2 Local control design

Fault-related shutdown

The disturbance output [Dstb] for an I/O function block is activated when the block recognizes a FAULT alarm (e.g., broken wire) or an OFFNORMAL alarm (e.g., exceeding a limit value).

The following figure shows how a valve and a pump are forcibly shut down or ramped up depending on the fault state.



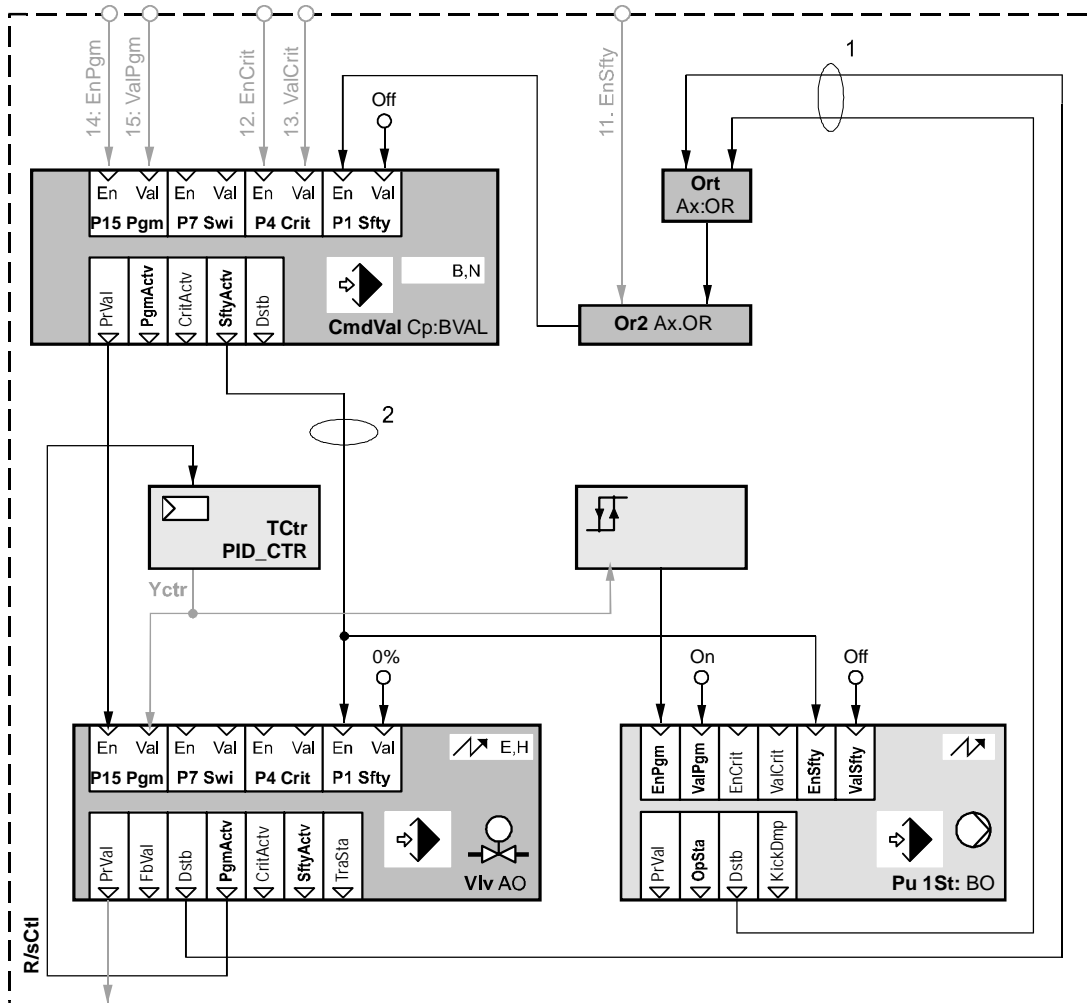
Example forced set-up


A limit value [HiLm] is defined for the temperature in block AI Temp. As soon as this threshold is reached, the output [Dstb] switches the valve via Enable [EnSfty] for the analog output value to 100%. At the same time, the pump is switched to off by Enable [EnSfty] for the Binary Output BO.

Example of fault-related shutdown

The block BI ThOvrd monitors the state of the pump's thermal switch. If the contact is triggered, the function block is activated based on the parameterized reference value [RefVal] for [Dstb] output. The pump is shutdown through Enable [EnSfty] of the Binary Output BO. The Binary Output BO further monitors the contact's feedback. In the event of a fault, where the feedback is interrupted, e.g., the block reports the fault and shuts down itself via the back wired output [Dstb]. The pump can only be switched on again only after the fault is eliminated and the alarm message is reset as required.

The following figure shows a local fault-related shutdown related to superposed plant control. The compound mapped here as an example was reduced to make is easier to recognize the structure of the local control.



 **Data Sink:** Receives Data
The Information is written by a Client with a certain priority or is read by the Function Unit.

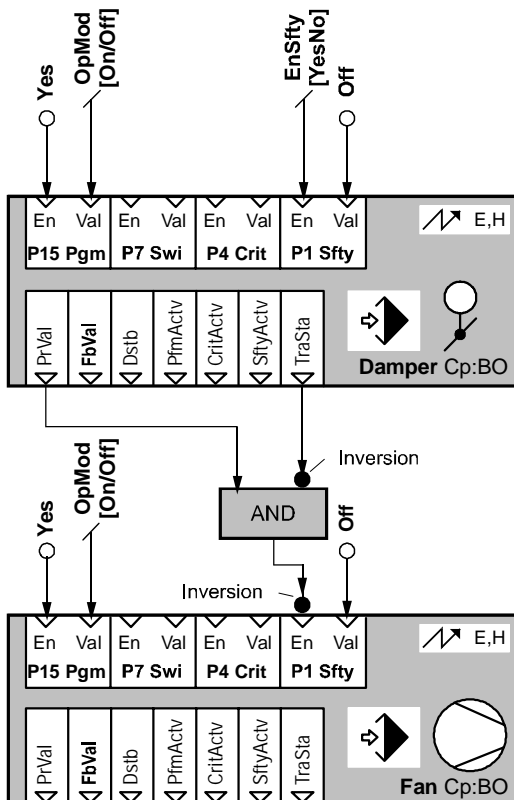
Local fault-related shutdown of the aggregate depicted here as an example is triggered as follows:

1. A fault is displayed at output [Dsb] when a component valve [Vlv] or pump [Pu1St] reports a fault (FAULT or OFFNORMAL). The signals revert to enable safety priority [EnSfty] for block BVAL (1). Fault-related shutdown of all components is triggered via the state output [SftyActv] (2).
2. You can also impact the safety shutdown of the components via the compound interface [I1 EnSfty].

The superposed plant control (not displayed here) can access the object directly via referencing since the block BVAL is mapped on BACnet and has a priority array [PrioArr]. As a result, plant control can also trigger a shutdown of the components by commanding the safety priority.

Interlocks

The following figure shows a solution where a fan is only enabled after the damper is completely open.



Local interlocks

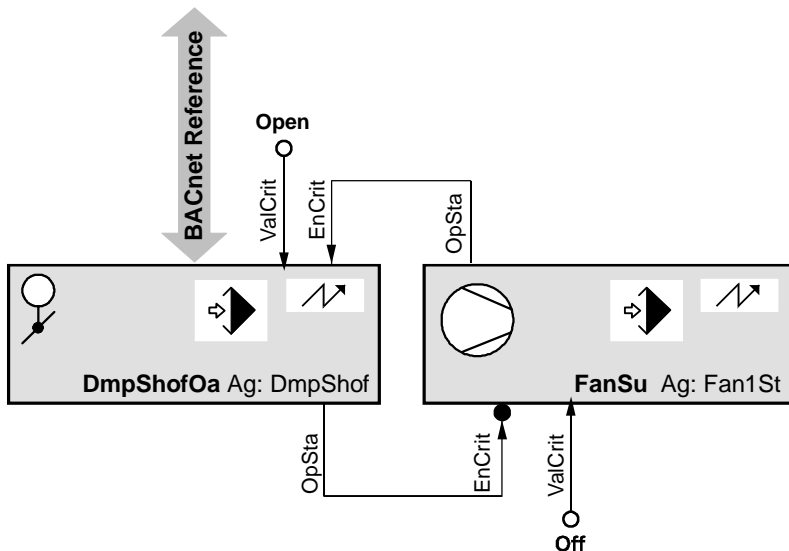
A command to ramp-up the plant [OpMod] = On, the damper output changes to [TraSta] = Yes, indicating that a transient state is now active, in other words, the damper is moving. This information is formed on the one hand from the parameterized damper run time [TbTiDly] and, on the other hand, from the feedback contact for the damper's mechanical stop.

The valve is blocked via input [EnSfty] as long as the damper is either blocked or moving, in other words, an intervention via the operator unit directly on the fan is prevented. When the transient state ends and the damper is open, the Enable [EnSfty] is cancelled and the fan switched on via the program value [ValPgm]. Enable of the program value [EnPgm] is a constant in this example.

Interlock among aggregates

The targeted interlocking is employed in a modified form from the superposed plant control. To allow, , plant control to access the fan during smoke extraction control, the interlock is not implemented by enabling the safety value [EnSfty], but rather by enabling the critical value [EnCrit].

The fan is set to Off by the damper via Enable [EnCrit] until the damper is fully open. The fan can only then start. The damper is held open via [EnCrit] as long as the fan is running to prevent a mistaken operation that could destroy the plant.



The operating state [OpSta] for both aggregates are formed within the compounds as illustrated in the previous example from the AND link for [PrVal] and [TraSta].

5.3 Superposed plant controls

Two blocks are available in Desigo for superposed plant control:

- CMD_CTL command control for sequence control
- PWR_CTL power control for stepped control

Both blocks are based on the standard BACnet Command Object. They have both tables that define the operating modes and switching response of the underlying aggregates. The commandable blocks in the aggregates must have a BACnet [PrioArr] to use the following output and value blocks: AO, BO, MO, AVAL, BVAL und MVAL.

PWR_CTL may only be communicated using the MVAL blocks based on the specialized task – controlling steps.

Referencing

Referencing is used exclusively for communications by the superposed control blocks with the output and value blocks in the aggregates to be commanded. The references are derived from the Technical Designation (TD) of the block. The reference is defined relative to the control block to the command block. The aggregate does not have to be in the same hierarchy; cross-plant communications is possible.

Example for a reference: B = \...\PreHcl'CmdVal

Where CmdVal is the designation for a BVAL object in the PreHcl aggregate. More than one block can be referenced for each aggregate.

As the project-specific root is not part of the address, the references do not need to be modified if the root changes. This simplifies project-specific name changes and the copying of library solutions into a project.

The references, that is, the technical designations with relative addresses are resolved in the controller at runtime. Any addressing errors will therefore only be apparent during runtime. The cause of the error can largely be eliminated, however, when parameterizing the controls blocks with the help of the Plant Control Editors.

The figure shows that the [PrioArr] can communicate directly with the referenced blocks. You can command switch and positioning values and enable them. A commanded command remains valid until the priority entry is enabled again. The control blocks automatically enables all commanded priorities, when the block commands the aggregates to the new plant operating mode. The entries for the [PrioArr] are deleted in the commanded blocks when restarting the PX controller, with the exception of local, manual interventions to priority 8.

Determining plant operating mode

A superposed plant control generally has different sources such as plant switch, scheduler program or important fault messages, from which the resulting plant operating mode must be determined.

The ENSEL_MS (Enable Selector Multistate) and ENSEL_BO (Enable Selector Boolean) blocks are available for evaluating the resulting plant operating mode in the firmware library of Desigo. As a rule, the block is placed

before plant control as illustrated in the following figure. All potential influences are interconnected, prioritized by importance on the block and the corresponding required plant operating mode is determined.

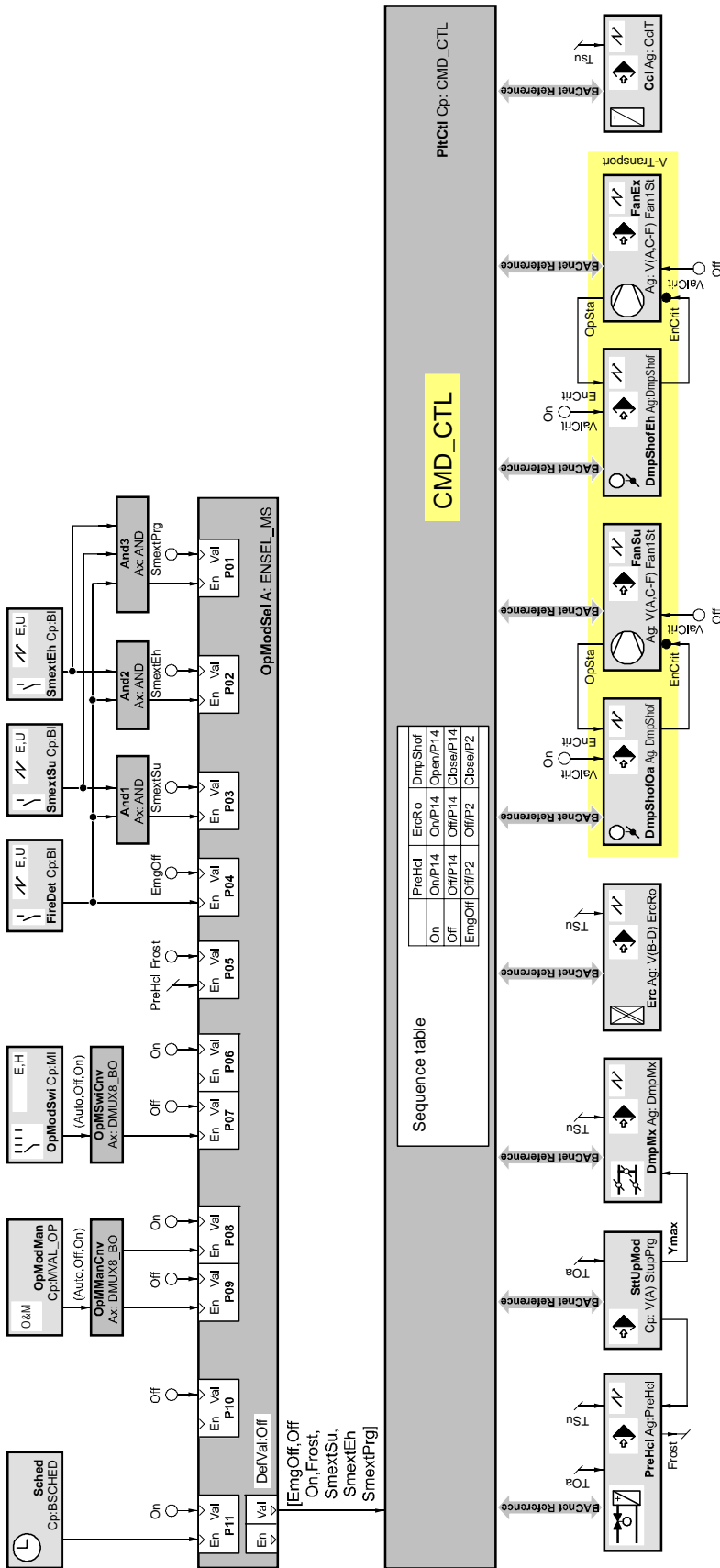
Example:

- A fire detector as a high priority (P04) and requires the plant operating mode EmergOff.
- The smoke extraction switch has the highest priority (P01) and demands plant operating mode smoke extraction.
- The scheduler has a low priority (P11) and demands plant operating modes Stage 1, Stage 2 and Off.

The output [Val] for ENSEL_MS now supplies the CMD_CTL the resulting plant operating mode for additional processing. It is important that the multistate enumerations for both blocks ENSEL_MS and CMD_CTL are the same. The multistate values are not text, but rather numbers based.

Superposed command control CMD_CTL

The command control CMD_CTL block is used primarily for the sequence control in the ventilation plants. The block makes it possible to sequentially switch on and off the aggregates. As it is implemented in a very general and flexible way, other fields of application are also conceivable, e.g., for refrigeration plants.



The block CMD_CTL controls and monitors output and value blocks mapped on BACnet. Communications is based on BACnet referencing rather than interconnections to optimize the costs of engineering. The following blocks can be used with CMD_CTL: AO, BO, MO and AVAL, BVAL and MVAL.

The sequence is determined in the CMD_CTL in a table. The command for the individual aggregates and the components can be determined based on the plant operating mode.

The main functionality of the block CMD_CTL is the sequential control of aggregates and components dependent on the preset plant operating mode [ValPgm]. For this purpose the switch-on sequence is defined by the order in the function table [FnctTb]. The switch-off sequence is the reverse of the switch-on sequence. Independent switch-on and switch-off sequences are not implemented in this block.

Switched on block can be monitored for their states. There is no monitoring of the OFF status.

Prior to switching on a block a test is made to see if the conditions for executing a command are given. The switch-on process is not even available for active switch on delay, minimum switch off times or a switch command with a higher command (e.g., a maintenance switch). This Look Ahead mechanism is described in greater detail in this chapter.

This block does not contain interlocks of individual functional units within aggregates. These are implemented locally via the data flow between the relevant blocks.

Plant Control Editor

The block parameters are set in the Plant Control Editor.

| Description | Mischluftklappe Fortluftklappe | Zuluftventilator Befehl | Abluftventilator Befehl | Temperaturregelung Befehlswert | Energierückgewinnung Befehlswert | Luftpühler Befehlswert |
|-----------------------------|-------------------------------------|-------------------------------------|-------------------------------------|-------------------------------------|-------------------------------------|-------------------------------------|
| Freigabe | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> |
| Objekt | B=\DmpMx\DmpEh | B=\FanSu\Cmd | B=\FanEx\Cmd | B=\TCl\CmdVal | B=\Erc\CmdVal | B=\Ccl\CmdVal |
| Betriebszustandsüberwachung | Zustand | Zustand | Zustand | Keine | Keine | Keine |
| Einschaltart | Verzögert | Sofort | Timeout | Sofort | Sofort | Sofort |
| Einschaltverzögerung | 00:00:30 | | 00:00:20 | | | |
| Ausschaltart | Sofort | Timeout | Sofort | Sofort | Sofort | Sofort |
| Ausschaltverzögerung | | 00:00:20 | | | | |
| Operating modes | | | | | | |
| Not-Aus | 0.00 | Aus | Aus | Aus | Aus | Aus |
| Priority | Personensicherheit, a | Personensicherheit, a | Personensicherheit, a | Personensicherheit, a | Personensicherheit, a | Personensicherheit, a |
| Aus | Not commanded | Aus | Aus | Aus | Aus | Aus |
| Priority | Spezifische Befehlsst | Spezifische Befehlsst | Spezifische Befehlsst | Spezifische Befehlsst | Spezifische Befehlsst | Spezifische Befehlsst |
| Stufe 1 | Not commanded | Ein | Ein | Ein | Ein | Ein |

The upper part of the dialog box serves primarily to provide a quick online overview of the present plant operating mode. You can also define exception value which become active as a plant operating mode during a plant fault.

The upper part of the table configures the sequences. The switch-on sequence of the objects, the monitoring mode and the switch on and off types for the sequence controllers can be defined here.

The lower part of the table is used to define the plant operating modes. You can define what command at what priority is command per plant operating mode for each sequence element.

The following priorities for commanding are available:

- Priority 2: Life safety, automatic
- Priority 5: Plant safety, automatic
- Priority 14: Specific command object
- Priority 16: System control

You can enable a command priority with the value Not command for plant operating modes where the local control is intended to assume control of the aggregate.

[DefVal] applies when the [PrioArr] for the corresponding block is empty, that is, not active recognition set, at that time.

Function workflows in CMD_CTL

A series of safety, monitoring and switch actions are conducted for each change to the plant operating mode in block CMD_CTL.

| Stage | Function | Action |
|-------|--------------------------|---|
| 1 | Safety function | Check AllLifeSafety plant operating modes. |
| 2 | Preview | Checks if the aggregates in question can be switched. |
| 3 | Abort sequence | Incomplete sequences are interrupted. |
| 4 | Reset sequence | Switch off unneeded aggregates. |
| 5 | Step-up sequence | Switch on the newly needed aggregates. |
| 6 | Monitor switch-on states | Start monitoring of countdown of delay period. |

Step 1: Safety function AllLifeSafety

If all switch commands for a given plant operating mode have the priority Life safety, it is referred to as the AllLifeSafety plant operating mode.

A pending AllLifeSafety plant operating mode in the [ValPgm] is executed immediately in all cases and maintained regardless of previously existing and newly occurring faults in the plant – human life takes precedence over plant safety.

If the AllLifeSafety mode includes switch-on commands, then the preset delay times (Delay and Timeout) will be observed. However, in the case of the Timeout setting, the switching sequence will continue even in the absence of any feedback signal. Interlocks cannot therefore be guaranteed, with the exception of local interlocks implemented via Priority 1 (life safety, manual).

Priority 1 (life safety, manual) cannot be overwritten in the AllLifeSafety.

Step 2: Preview Look Ahead

Before changing to a different plant operating mode, in which referenced blocks are to be enabled, block CMD_CTL checks to ensure that all the aggregates can actually be enabled. For this purpose, the entries in the priority array [PrioArr] for the switching sequence blocks are checked in advance. If switch commands of a higher priority are found to be active (e.g., a minimum switch-off time or the OFF-command of a repair switch), then CMD_CTL waits to implement the new plant operating mode until the full switching sequence can be implemented. Only referenced blocks, for which a switch-on command exists in the new plant operating mode, are checked, and only if the operating-state monitoring feature has been enabled.

The following priorities are checked:

- Priority 1 [EnSfty/ValSfty], life safety, manual.
- Priority 7 [EnSwi/ValSwi], manual operation, e.g., manual switch.
- Priority 8 [EnOp/ValOp], manual operation, operating unit.
- Priority 6 [TiMinOff], minimum switch off time.

Priority 6 is checked only for a switch on command to determine whether the aggregate is still within the minimum switch off time. In this case, it waits until the switch off time expires and only then switches on.

There is no Look Ahead for Designo 7.

Priority 4 (plant safety, manual [EnCrit/ValCrit]) is not considered during the check, since local mutual locking via data flow interconnection, such as depicted in the figure *Cross-aggregate interlocking of damper/fan*, would change this value during the switch-on process.

The present operating mode remains until it is certain that all impacted aggregates with active operating state supervision can be switched to the new set state. A process alarm is triggered in CMD_CTL of a monitored block is not switched on. The exception value [EcptVal] is active as the new plant operating mode in this case. The online diagnostics for the Plant Control Editors determines which element is the cause of the fault.

Step 3: Abort sequence

On-going switch sequences are aborted when delay times are still active. Exception: An alarm is generated when a fault occurs as part of internal monitoring of the block. The demanded plant operating mode is determined in this case by the exception value [EcptVal]. If the switch sequence is active, but not completed, it is NOT aborted, but rather is completed.

Step 4: Ramp-down sequence

The ramp-down sequence is started first for the new plant operating mode. This shuts down all aggregates that must be switched off per the new plant operating mode. The shut down takes place in the table sequence from right to left, in other words, the last aggregate in the switch sequence is the shut down first. The parameterized

times for the time off delay are active during ramp down to off. The time off delay can be activated using a fixed delay time or a maximum timeout or deactivated using the immediate option. The length of the delay for timeout depends on the switch off state of the monitored sequence elements. Transition to the next sequence occurs as soon as it reports switched off, that is, the process value of the block [PrVal] = Off. It switches after the timeout time expires when the shut-down message is not sent.

If a sequence element with a life-safety or plant-safety priority is switched off, the preset delay times will be ignored.

Step 5: Step-up sequence

The step-up sequence is then started for the new plant operating mode. The remaining aggregates are switched on per the data in the function table. The switch on takes place in the table sequence from left to right, in other words, the first aggregate in the switch sequence is the switched on first.

The parameterized times for the time on delay is active during step-up.

The step-up delay can be activated using a fixed delay time or a maximum timeout or deactivated using the immediate option. The length of the delay for timeout depends on the switch on state of the monitored sequence elements. Transition to the next sequence occurs as soon as it reports switched on, that is, the process value of the block [PrVal] <> Off. It switches after the timeout time expires when the switch on message is not sent.

When a sequence element with a life-safety or plant-safety priority is switched on, the preset delay times will take effect first.

Step 6: Monitoring switch on state

A process alarm (off normal) is generated when the monitored aggregate is not switched on after the sequence delay time expires.

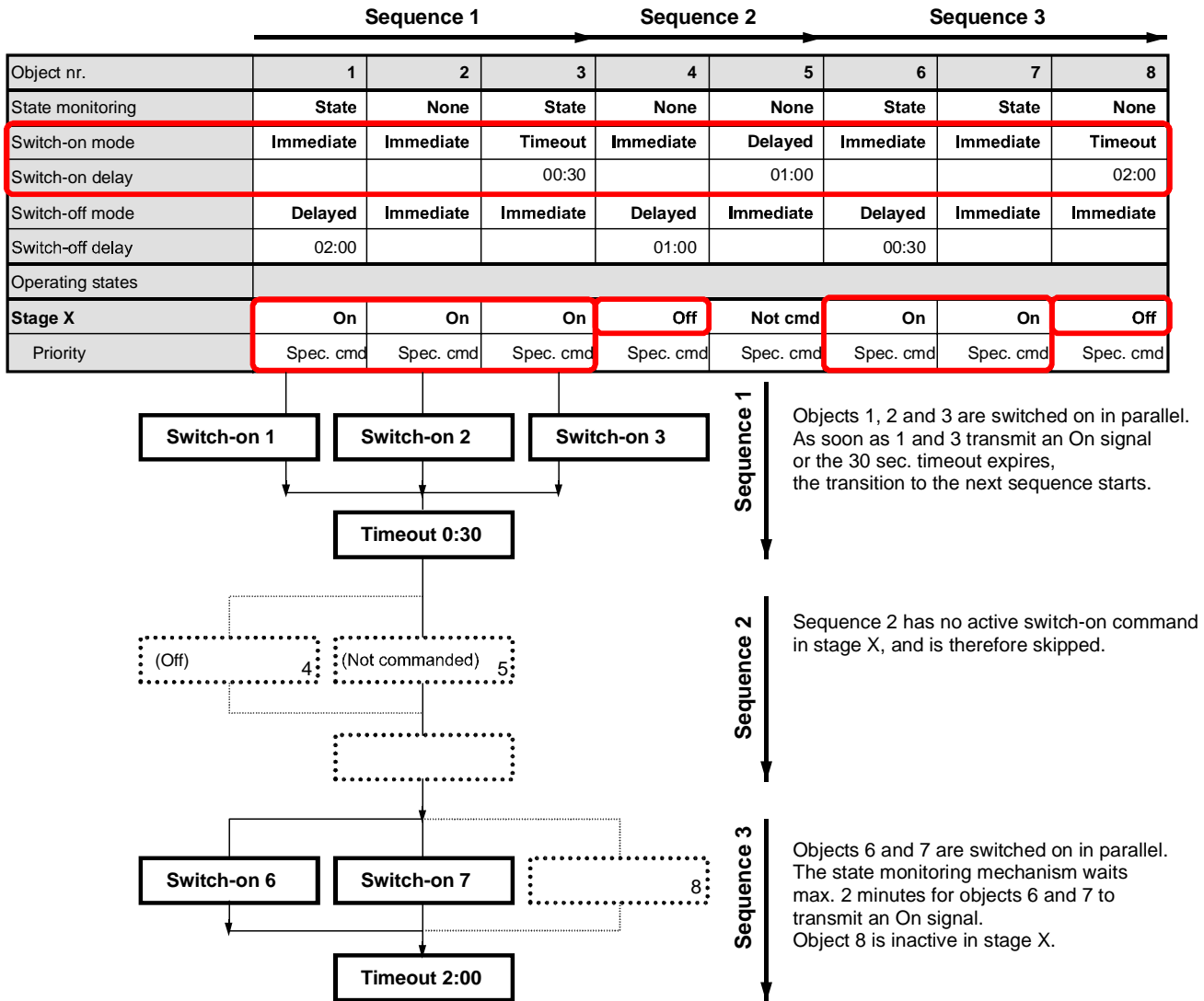
The current switch sequence is immediately aborted when the current plant operating mode is not AllLifeSafety and the exception value [EcptVal] is selected as the operating mode.

If, however, the exception value [EcptVal] is already the plant operating mode, the switch sequence is not aborted and the plant operating mode does not change.

Switch on aggregates

The following figure shows the switch response and monitoring mechanism for block CMD_CTL.

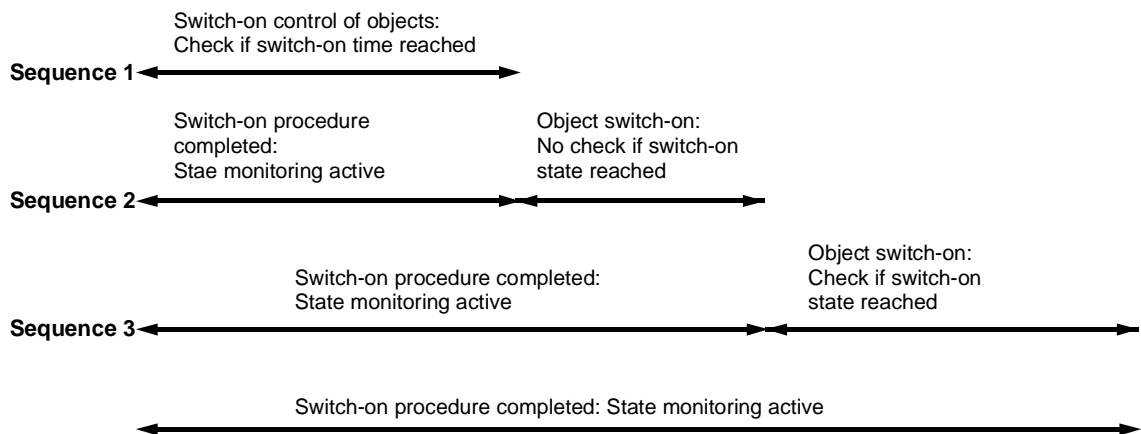
The system initially checks if the new plant operating mode is an AllLifeSafety mode. The Look Ahead check takes place in the second step, followed by the check and abort of on-going sequences. The next step is to run the shut-down series, where objects 8 and 4 are switched off to the extent they have not yet be shut down. The sequences are then switched on one after the other in the follow-on switch on series.



The set time (delay or timeout) marks a switch on or off sequence that may consist of one or more objects. The times apply for the entire sequence and take effect, when a switch on or step up command or a switch off or ramp down command is demanded.

Switch on occurs in parallel per sequence. A check of the switch on state occurs only in the switch on type timeout. The next sequence is only started after either all monitored objects report a switched on state or the timeout period expires. Operating state monitoring of the objects for monitoring, as depicted in the following figure, only become active after the step-up process for a sequence is completed.

| | Sequence 1 | | | Sequence 2 | | Sequence 3 | | |
|------------------|------------|-----------|-----------|------------|-----------|------------|-----------|-----------|
| Object nr. | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
| State monitoring | State | None | State | None | None | State | State | None |
| Switch-on mode | Immediate | Immediate | Timeout | Immediate | Delayed | Immediate | Immediate | Timeout |
| Switch-on delay | | | 00:30 | | 01:00 | | | 02:00 |
| Switch-off mode | Delayed | Immediate | Immediate | Delayed | Immediate | Delayed | Immediate | Immediate |
| Switch-off delay | 02:00 | | | 01:00 | | 00:30 | | |
| Operating states | | | | | | | | |
| Stage X | On | On | On | Off | Not cmd | On | On | Off |
| Priority | Spec. cmd | Spec. cmd | Spec. cmd | Spec. cmd | Spec. cmd | Spec. cmd | Spec. cmd | Spec. cmd |



Operating status monitoring is optional and monitors only blocks in a Switched on state. If a referenced block is found to be switched off during active operating state monitoring, but that the block should have been in the state Switched on, a process alarm is generated and the plant operating mode changes to exception value [EcptVal].

The momentary alarm state is visible from the state flag [StaFlg].

| | Sequence 1 | | | Sequence 2 | | Sequence 3 | | |
|------------------|------------|-----------|-----------|------------|-----------|------------|-----------|-----------|
| Object nr. | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
| State monitoring | State | None | State | None | None | State | State | None |
| Switch-on mode | Immediate | Immediate | Timeout | Immediate | Delayed | Immediate | Immediate | Timeout |
| Switch-on delay | | | 00:30 | | 01:00 | | | 02:00 |
| Switch-off mode | Delayed | Immediate | Immediate | Delayed | Immediate | Delayed | Immediate | Immediate |
| Switch-off delay | 02:00 | | | 01:00 | | 00:30 | | |
| Operating states | | | | | | | | |
| Stage X | On | On | On | Off | Not cmd | On | On | Off |
| Priority | Spec. cmd | Spec. cmd | Spec. cmd | Spec. cmd | Spec. cmd | Spec. cmd | Spec. cmd | Spec. cmd |

Monitoring is active from the point when the corresponding sequence successfully completes the switch on process, that is, the process value for block [PrVal] is not equal to Off and the transient state is completed [TraSta] = No.

The [PrVal] of the block will be monitored. Hence, only those events which affect [PrVal] can be detected, that is:

- Local fault shut down using interconnection of fault [Dstb] to enable safety, manual [EnSfty].
- Local shut down of the block in a higher priority application program.
- Switch-off by manual operation of the output module if the I/O module returns the manual setting value.
- Block switched off via HMI operation or manual switch in control panel

Command control is only in a position to recognize fault-related deviations and act accordingly when the interconnection of all relevant faults [Dstb] occur locally on a monitored output of value block to [EnSfty]. Its default value [DefVal] becomes the process value [PrVal], if a referenced output or value block is out of service [OoSrv]. The state monitoring of the plant cannot operate correctly, since [PrVal] no longer reflects the actual state of the aggregate.

To reduce the frequency with which aggregates are switched on and off, it is possible to define a minimum switch-off time [TiOffMin] in the aggregates. The look-ahead mechanism in the CMD_CTL block prevents the switching of the whole sequence if the minimum off-time in one aggregate with active state-monitoring has not yet expired. The output [TraSta] shows the transitional state and [PrVal] remains unchanged, at the last value. The new plant operating mode will be implemented only when all the aggregates to be enabled in the switching sequence can actually be enabled.

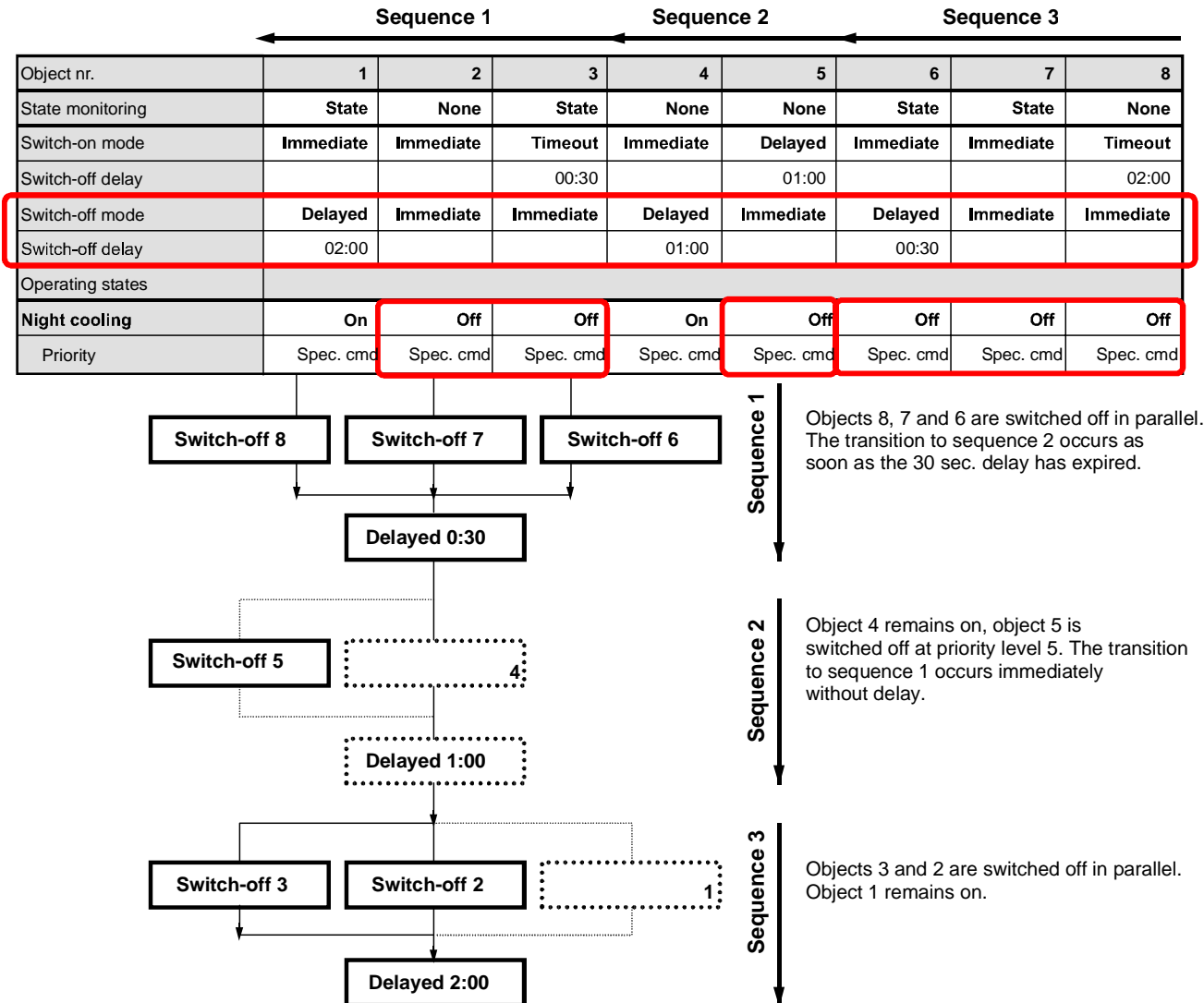
A minimum off-time should always be set for aggregates incorporating a rotating mass (e.g., fans).

Operating mode changeover

The following figure shows a changeover from operating mode Stage Y to Night cooling.

All objects were switched on in Stage Y. During the changeover to Night Cooling, the system initially checks whether the new plant operating mode is an AllLifeSafety mode. The Look Ahead check takes place in the second step; followed by the check and abort of on-going sequences.

In the next step, the switch off series is conducted where the sequence elements of switch off sequence 1 are switched off in parallel. It transitions to the second sequence after the delay time expires. Object 5 is commanded to Off with plant safety, priority 5. For plant safety or life safety (priority 2), the delay times or timeouts have no effect. The transition to switch-off sequence is immediately since object 4 is already switched on.



Objects 3 and 2 are switched at the same time as object 5. Object 1 remains switched on.

Alarm management

Block CMD_CTL is alarmable and differentiates process from system alarms.

A process alarm occurs, when:

- One of the monitored aggregates is not switched on.
- One of the referenced aggregates cannot be switched on.

The exception value [EcptVal] becomes the present plant operating mode as a reaction to a process alarm. In addition, an alarm is sent.

A system alarm occurs for the following configuration efforts:

- A referenced aggregate is not available.
- A referenced aggregate is not a commandable object.
- Impermissible priorities are used (priorities 2, 5, 14, 16 are allowed).
- [ValPgm] or [EcptVal] are outside the permissible range.
- The referenced aggregates have a different number of operating modes.

The command control attempts for a system alarm to enable all referenced blocks for local control. The four commandable priorities are commanded – in other words enabled to Not commanded: Life safety (2), plant safety (4), specific command control (14) and system control (16).

The response of the block to an alarm can be defined. The following mechanisms have been incorporated to prevent hunting in the plant.

- Basic and standard: When the block goes into alarm, the exception value [EcptVal] is switched. When all the aggregates are ready for switching again, CMD_CTL automatically tries to implement the present plant operating mode [PrVal]. If all the aggregates are ready for direct switching immediately after implementation of the exception value [EcptVal], hunting is likely to occur. In this case, CMD_CTL prevents any further switch-on attempt, and the required plant operating mode [PrVal] must be changed.
- Extended: When the block goes into alarm, the exception value [EcptVal] is switched. The alarm has to be reset by the user, and there is therefore no risk of hunting.

The block is not alarmable for Designo 7.

Out of service

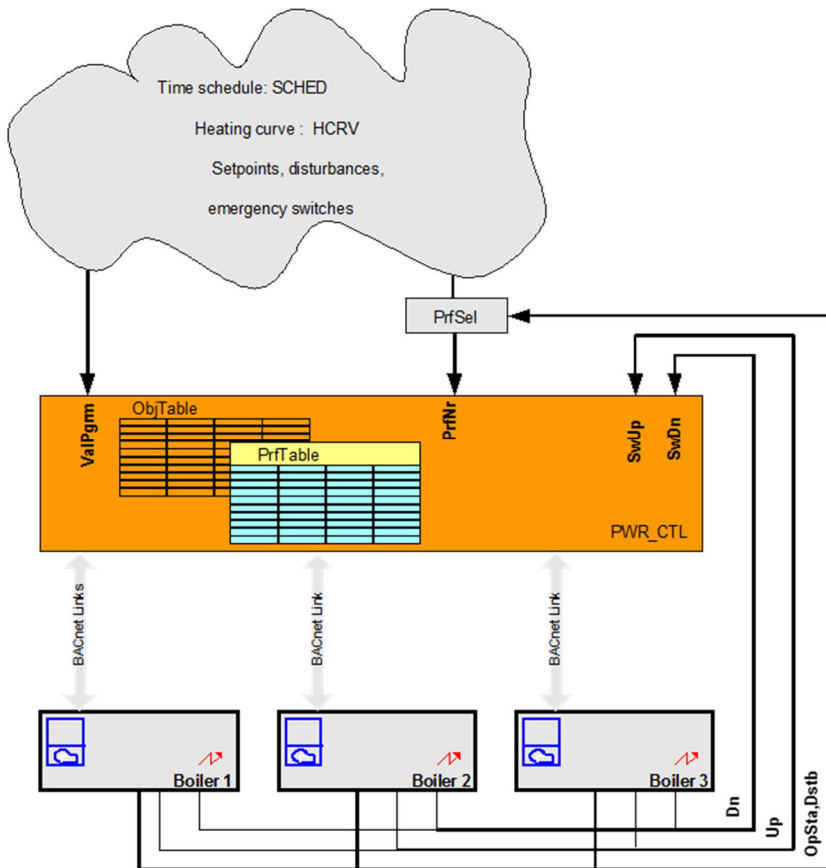
The block can be taken out of commissioning using [OoServ]. The following occurs when switching [OoServ] to On:

- Immediately abort of switch on and off sequences and monitoring.
- All objects are commanded with a release of the priorities: Life safety (2), plant safety (4), specific command control (14) and system control (16)

Superposed power control PWR_CTL

The power control function block PWR_CTL is used for control and monitoring of the performance of a number of energy producers (multiple boiler systems, refrigeration machines, etc.). As is the case for command control CMD_CTL, the data is exchanged bilaterally between power control and the individual energy producers (boiler, refrigeration aggregate, among others), via referencing. Since the energy producers are generally implemented in the form of logical aggregates, and contain local logic, the PWR_CTL block communicates only with MVAL blocks.

The control strategy is based on the use of tables and is designed for multi-stage energy producers. Additional energy-producer stages are connected or disconnected in accordance with the actual power demand. For modulating energy producers, a stepped output is converted into a proportional output within the aggregate. This makes it possible to handle the full power range (0...100%) in one stage, or to divide the power range into several stages (e.g., Stage 1: 0...20%; Stage 2: 20...40%; etc.).



Plant Control Editor

The block parameters are set using the Plant Control Editor.

| Beschreibung | Boiler 1 | Boiler 2 | Boiler 3 |
|--------------------------|-------------------------------------|-------------------------------------|-------------------------------------|
| Freigabe | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> |
| Objekt | B='Bo1'CmdVal | B='Bo2'CmdVal | B='Bo3'CmdVal |
| Einschaltverzögerung | 00:15:00 | 00:10:00 | 00:10:00 |
| Ausschaltverzögerung | 00:00:30 | 00:00:30 | 00:01:00 |
| Hochschalt-Verzögerung | 00:03:00 | 00:03:00 | 00:03:00 |
| Zurückschalt-Verzögerung | 00:00:30 | 00:00:30 | 00:00:30 |
| Leistung Einschaltstufe | 50 | 100 | 100 |
| Leistung nächste Stufe | 100 | 50 | 100 |

The upper part of the dialog box serves primarily to provide a quick online overview of the block. The maximum power controlled by the block is set with the maximum power parameter [MaxPwr]. The value must be greater than 0 kW in order for the block to work. Any changes in this limit value have a direct effect in online mode. If no limit value is required, the maximum power must be set at an appropriately high value.

The Aggregates tab is used to set the control variables of the aggregates (boiler, refrigeration machine).

- Enable: Activation/deactivation of an entry if they are not released, aggregates in the Profile table will be ignored.
- Command object reference: Reference (relative addressing) to multistate value blocks [MVAL] of the relevant energy producer. During the configuration process, all MVAL blocks at the same and at lower hierarchical levels are displayed.
- Aggregate description: The reference to the value object provides access to (and hence, knowledge of) all information in a special dialog box via the referenced object in the control command.
- Switch-on delay: Delay time when switching from OFF to Stage 1.
- Switch-off delay: Delay time when switching from Stage n to OFF.
- Step-up delay: Delay time when switching from Stage n up to Stage n+1.

- Step-down delay: Delay time when switching down from Stage n to Stage n–1.
- Switch-on stage power: Power in [kW] at the lowest (that is, first) stage.
- Next power stage: Additional power at the next stage(s) in [kW].

The control sequences for the aggregates (boiler, refrigeration machine) are defined under the Aggregates tab. Each profile describes the order in which the energy producers are to be switched and the maximum stage in each case. A total of 8 profiles each with 15 sequence entries can be defined.

Plant Control Editor - Site:HGen'PwrCtl [Leistungssteuerung]

Profil: Aktion:

Maximale Leistung: Aktuelle Leistung:

Aggregate Profil

| | Beschreibung | Profil 1 | Profil 2 | Profil 3 |
|---|------------------|----------|----------|----------|
| 1 | Objekt | Boiler 1 | Boiler 3 | Boiler 3 |
| | Stufenbegrenzung | Stufe 2 | Stufe 2 | Stufe 1 |
| | Regelungsart | Freigabe | Freigabe | Freigabe |
| 2 | Objekt | Boiler 2 | Boiler 3 | Boiler 1 |
| | Stufenbegrenzung | Stufe 4 | Stufe 1 | Stufe 2 |
| | Regelungsart | Freigabe | Freigabe | Fest |
| 3 | Objekt | Boiler 3 | Boiler 1 | Boiler 2 |
| | Stufenbegrenzung | Stufe 3 | Stufe 2 | Stufe 1 |
| | Regelungsart | Freigabe | Freigabe | Freigabe |

The active profile table is defined by entering the profile number [PrfNr] as an input parameter, or by selecting it from the Profile dropdown list in the Plant Control Editor. This input parameter can be interconnected, so that the profile can be changed as a function of other events (faults, summer mode, boiler operating hours, etc.). If the profile is changed during operation, the power output [PrPwr] is switched in accordance with the power profile in the new profile table.

The profile definition determines the order in which individual aggregates are to be switched on or off. The following information must be entered for every sequence entry:

- Object: Selected from the previously referenced aggregates.
- Stage limitation: Limit up to which the aggregate may be enabled.
- Control type: Specifies whether the enabled stages are to be switched permanently or released to the control system.
 - Fixed: The total power provided by a given switch stage is switched on or off permanently. This option can be used, e.g., for a specific base load which is required to be present at all times. The command is implemented with Priority 14.
 - Enable: The power actually required from the released switch stage is determined by local control of the aggregates. The command is implemented with Priority 16.

For each sequence step, the function block only ever releases the last aggregate marked Release to the control system. It displays this released object [RlsObj], with the released object stage [RlsObjSt], at the output. All other aggregates are fixed at the released stage value. If none of the aggregates is marked Release, the aggregate of the current sequence step is released to the control system.

On/Off switching of PWR_CTL

When PWR_CTL is switched on [ValPgm = On], the first step in the sequence of the current profile is executed immediately. In this case, the switch-on delay is not valid. If the trigger for default power is on [PwrTrg = On], the aggregate is switched directly to the default power level [DefPwr].

A switch-off command [ValPgm = Off], disables all the energy producers defined in the profile table with Priority 14.

Out of service

If the PWR_CTL function is taken out of service [OoServ = On], then all referenced aggregates are switched OFF with Priority 14, without taking account of delay times. The monitoring of the aggregates is disabled.

Demand signals

The current power demand is determined locally in the energy producers. In the event of a power deficit or surplus, the aggregate will send the appropriate demand signal to the PWR_CTL block. The demand signal from the aggregate can be generated, e.g., on the basis of the boiler setpoint deviation and the primary flow. The demand signals of the separate aggregates are combined and transmitted to the [StepUp] or [StepDn] input of the PWR_CTL block. After expiry of the relevant delay times, the block executes the appropriate sequence step to increase or reduce the power, as necessary.

When both [StepUp] and [StepDn] demand signals are present simultaneously, [StepDn] takes priority.

Direct switching of a load

In cases where the power is to be increased or decreased without observing the delay times, the default-power trigger input [PwrTrg] can be used to switch to a defined default power level [DefPwr]. From the current profile, and taking account of the current power output, the PWR_CTL block determines the sequence steps required to cover the power demand and implements them directly.

Power display

The block has two outputs at which it displays the current total power of the energy producers. This consists firstly of the controlled power output [CtldPwr]. This output represents the total power switched by the PWR_CTL block.

The other output, the present power output [PrPwr], shows the additional power output of energy producers that are not directly switched by PWR_CTL. To do this, PWR_CTL evaluates the priority array [PrioArr] of the MVAL blocks. In this way it can detect, e.g., that an energy producer has been switched manually [Prio8] to a given stage.

Configuration error

The entries in the two configuration tables are checked cyclically for validity.

- A fault alarm is generated under the following circumstances:
- Aggregates no longer accessible from PWR_CTL, owing, e.g., to retrospective modifications to the technical hierarchy, affecting the references of the energy producers
- Retrospective changes to the stage-limit value in the aggregate, making the value configured in the profile table too high
- No multistate value object
- Reference block no longer available: e.g., deleted with delta download
- Several references to the same block
- Empty profile table

In the event of a fault alarm, all aggregates still accessible by PWR_CTL are switched OFF permanently.

Alarm management

The PWR_CTL block in the system is an alarm-generating block with a configurable alarm class [AlmCl] and alarm function [AlmFunct].

An Offnormal process alarm is generated:

- When the step-up demand signal [StepUp] persists for longer than the monitoring time deviation [TiMonDev], and there are no further sequence steps to increase the power.
- When the step-up demand signal [StepUp] persists for longer than the monitoring time deviation [TiMonDev] plus the step-up delay time of the next sequence step, AND a step-up would cause the maximum power limit [MaxPwr] to be exceeded.

The process alarm is reset to normal:

- When a sequence step with an increase in power becomes possible again. Another sequence step with an increase in power becomes possible when the [MaxPwr] limit will no longer be exceeded, or when a further sequence step with a power increase is present.
- When there is no further [StepUp] demand signal

The text of process alarms can be defined to suit customer requirements.

Switching alternatives

Various switching alternatives can be defined by entries in the profile table. Note that where one or more step-up sequence steps (intended to increase the power) would, in practice, result in a drop in the power output, all steps in the step-up sequence are enabled automatically until a sequence entry is reached, at which the power output actually does increase as required. See also the following example.

Plant Control Editor - Site:HGen'PwrCtl [Leistungssteuerung]

Profil: Aktion:

Maximale Leistung: Aktuelle Leistung:

Aggregate | Profil

| Beschreibung | Boiler 1 | Boiler 2 | Boiler 3 |
|--------------------------|-------------------------------------|-------------------------------------|-------------------------------------|
| Freigabe | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> |
| Objekt | B='Bo1'CmdVal | B='Bo2'CmdVal | B='Bo3'CmdVal |
| Einschaltverzögerung | 00:15:00 | 00:10:00 | 00:10:00 |
| Ausschaltverzögerung | 00:00:30 | 00:00:30 | 00:01:00 |
| Hochschalt-Verzögerung | 00:03:00 | 00:03:00 | 00:03:00 |
| Zurückschalt-Verzögerung | 00:00:30 | 00:00:30 | 00:00:30 |
| Leistung Einschaltstufe | 50 | 100 | 100 |
| Leistung nächste Stufe | 100 | 50 | 100 |

The power data in the object table and the sequence entries in the profile table in Figure *Example of aggregate table* together give the power profile illustrated in Figure *Example of profile entries with a drop in power (Profile 2)*.

Profile 1

In the main application of the PWR_CTL function, a new energy producer is added for each sequence entry in the profile table. For this purpose, an aggregate only needs to be entered in the sequence table once.

In the event of a power demand, which the boiler transmits to the PWR_CTL function in the form of the [StepUp] demand signal, a further boiler stage / sequence step is enabled when the step-up delay has expired. When a boiler reaches the stage limit, the function switches to the next boiler or boiler stage after expiry of the switch-on delay. The last-enabled boiler stage is released to local power control, while all other boilers are fixed at their current power output.

If the power needs to be reduced, this is transmitted to the PWR_CTL function via the [StepDn] demand signal. The sequence steps are then executed in reverse order, with the defined switch-off and step-down delay times.

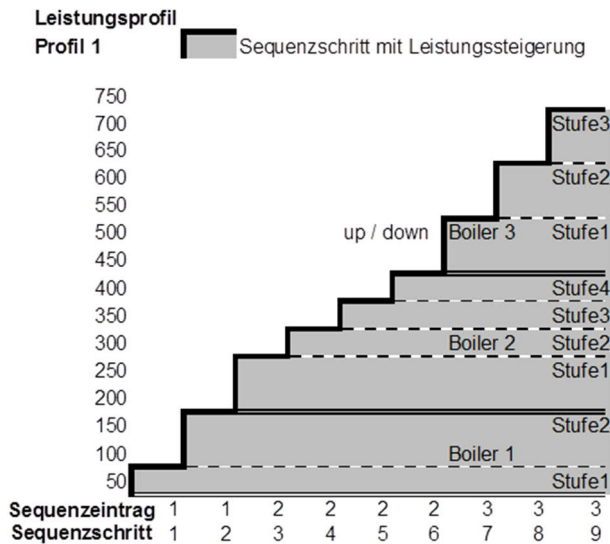
Plant Control Editor - Site:HGen'PwrCtl [Leistungssteuerung]

Profil:

Maximale Leistung:

Aggregate | Profil

| | Beschreibung | Profil 1 |
|---|------------------|----------|
| 1 | Objekt | Boiler 1 |
| | Stufenbegrenzung | Stufe 2 |
| | Regelungsart | Freigabe |
| 2 | Objekt | Boiler 2 |
| | Stufenbegrenzung | Stufe 4 |
| | Regelungsart | Freigabe |
| 3 | Objekt | Boiler 3 |
| | Stufenbegrenzung | Stufe 3 |
| | Regelungsart | Freigabe |



Profile 2

Profile 2 shows that the order in which boiler stages are to be enabled has been changed, and that sequences which will cause a drop in the power output have been defined in the power profile. In the example illustrated, Boiler 3, which is currently delivering 200 kW, is switched OFF via sequence entry 2. Boiler 1, which could achieve a power output of 150 kW with its enabled stages, is defined as the next object in the sequence. This results in a drop in the power output, causing the function block to enable all sequence steps automatically until an actual increase in power is achieved.

In sequence entry 4, Boiler 2 is enabled up to stage 2, giving a further 150 kW output. Boilers 1 and 2 are thus enabled simultaneously up to stage 2, to prevent a drop in power. The effective delay time for the simultaneous switching process is determined the maximum delay time of the boilers concerned. Since it is Boiler 1 which has the longest delay (15 minutes), the simultaneous switching operation will be delayed for this period of time.

Sequence entry 5 would again result in a drop in performance, because stage 2 of Boiler 2 is no longer enabled. The block therefore switches straight to sequence entry 6, enabling Boiler 3 to compensate for the power deficit. In this case the effective switch-on time is based on the switch-on delay for Boiler 3 (10 minutes).

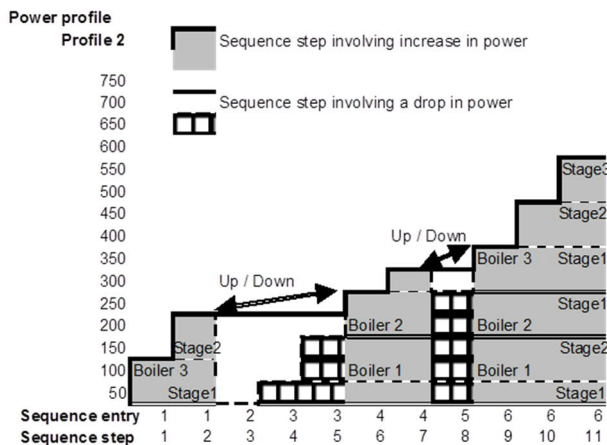
Plant Control Editor - Site:HGenPwrCtl [Leistungssteuerung]

Profil: Aktion

Maximale Leistung: Aktuelle Leist

Aggregate Profil

| | Beschreibung | Profil 1 | Profil 2 |
|---|------------------|----------|----------|
| 1 | Objekt | Boiler 1 | Boiler 3 |
| | Stufenbegrenzung | Stufe 2 | Stufe 2 |
| | Regelungsart | Freigabe | Freigabe |
| 2 | Objekt | Boiler 2 | Boiler 3 |
| | Stufenbegrenzung | Stufe 4 | Aus |
| | Regelungsart | Freigabe | Freigabe |
| 3 | Objekt | Boiler 3 | Boiler 1 |
| | Stufenbegrenzung | Stufe 3 | Stufe 2 |
| | Regelungsart | Freigabe | Freigabe |
| 4 | Objekt | | Boiler 2 |
| | Stufenbegrenzung | | Stufe 2 |
| | Regelungsart | | Freigabe |
| 5 | Objekt | | Boiler 2 |
| | Stufenbegrenzung | | Stufe 1 |
| | Regelungsart | | Freigabe |
| 6 | Objekt | | Boiler 3 |
| | Stufenbegrenzung | | Stufe 3 |
| | Regelungsart | | Freigabe |



Online diagnostics

A diagnostics screen for the PWR_CTL block is available online in Xworks Plus (XWP).



The following states are displayed:

- Present value: Operating state at the block output pin [PrVal]
- Action: Transient state [TraSta] depending on actual switching conditions: Up, down or hold
- Present power: Value at the block output pin [PrPwr]
- Status flag: In accordance with the BACnet definition, the value of [StaFlg] is always Overridden. Alarms may also be displayed here.
- Released aggregate or stage: This shows the current sequence entry, the released object [RIsObj] and the released object stage
- Last alarm/event message: Value at the block output pin [LstMsg]

5.4 Closed-loop control strategy

Controller types

For closed-loop control purposes, two controller blocks are provided in the Desigo system, which cover the majority of requirements:

- [PID_CTR]
- [CAS_CTR]

PID_CTR stand-alone controller – Sequence controller

The PID_CTR block is used as:

- A universal stand-alone PID controller
- A universal PID controller with external tracking
- An individual sequence-controller element in a sequence controller or sequence cascade controller

The PID_CTR block integrates the following functions:

- Can be programmed for P, PI, PID or PD control action
- Gain, integral action and derivative action can be programmed individually
- Proportional control output with minimum and maximum limit control
- Programmable gain factor
- Programmable neutral zone
- Programmable offset (for P and PD controllers)

- Programmable initial integrator value (for PI or PID controllers)
- Programmable runtime for control variable (0 – 100%, 100 – 0%) and positioning speed
- Type of operation (direct acting or reverse acting) can be selected

A sequence controller can be implemented by interconnecting several PID_CTR blocks. The sequence linker SEQLINK can also be used, where appropriate. The only function of this block is to enable individual sequence elements to be deleted without the need to create new connections.

CAS_CTR cascade controller

The PID_CTR block is used:

- As the master controller in a sequence cascade control configuration (e.g., room/supply air cascade).
- In temperature and humidity control loops

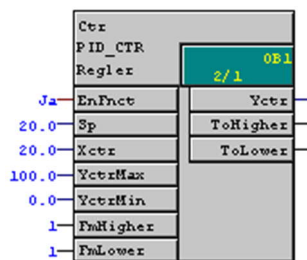
The following functions are integrated in the CAS_CTR block:

- Can be programmed for P, PI, PID or PD control action
- Proportional controller output with minimum and maximum limit control
- Setpoints for heating and cooling sequences, and for energy recovery
- Setpoint depending on type of operation, for energy recovery
- Initialization of integrator (initial value)

Universal PID controller

The PID_CTR block can be used as a universal stand-alone controller in a plant for the control of any control variables, e.g.:

- Temperature, temperature differential
- Pressure, pressure differential
- Velocity
- Absolute humidity, relative humidity



Control action

The PID_CTR block can be configured as a P, PI or PID controller. The following parameter settings are used to define the control action:

- Gain [Gain]
- Integral action time [Tn]
- Derivative action time [Tv]

As an option, the gain [Gain] can be influenced with the [GainFac] input. It can be useful to correct the gain factor in this way when controlling outside air dampers, e.g., as the effect of the damper positions can depend on the outside air temperature. The correction factor is defined with the gain scheduling block ADAGAIN.

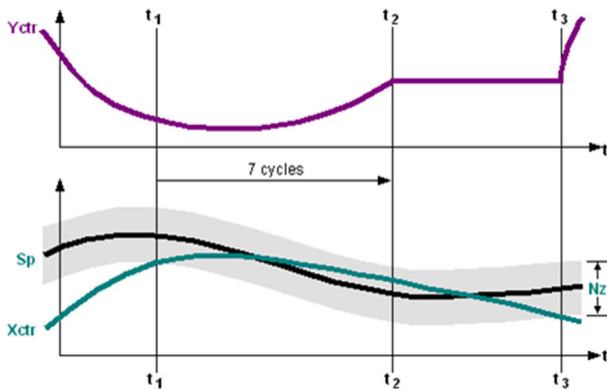
The actuator runtime can be set. Specifying the actual actuator run-times makes it possible to tune the controller more accurately to the actuator concerned, so improving the control quality of the control system.

Correcting range

The correcting range is limited by specifying the minimum and maximum output variable. In this process, the minimum of the two values is always set as the maximum value. In other words, the maximum value may be below the minimum value; there is no need to update the minimum value.

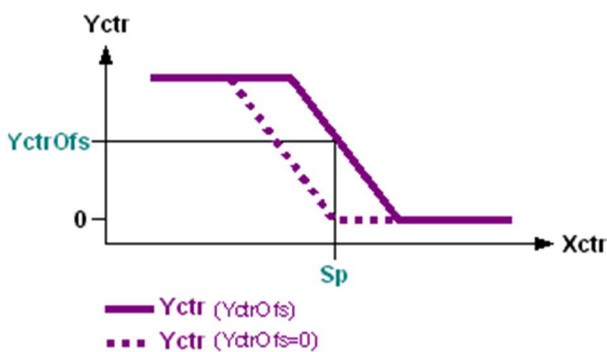
Neutral zone [Nz]

[Nz] is a zone on either side of the setpoint, within which the controller does not respond. As soon as the difference between the setpoint [Sp] and the measured value [Xctl] is less than half of the [Nz], the output is driven for a further 7 cycles, so that the measured value [Xctl] is as close as possible to the middle of the [Nz]. The output signal [Yctr] then remains constant. The output signal is only re-adjusted when the parameters move outside the [Nz] again.



P/PD controller

If the PID_CTR block is configured as a P-controller or PD-controller, a calibration point (Offset) [YctrOfs] can be specified, e.g., the P-controller can be calibrated so that the set point is maintained with a 50% load. With a 0% or 100% load, the P-deviation is then half the amplitude of the proportional range [Gain].



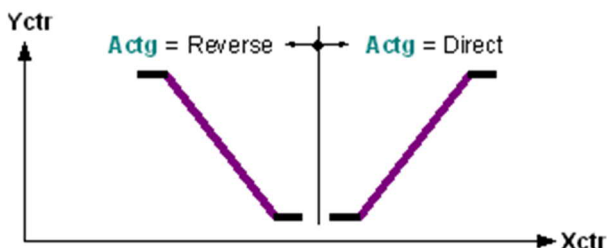
Tracking [Track]

[Track] is used, e.g., where the PI(D) controller, operates as a limit controller, e.g., acting on a valve or actuator via an intermediary minimum or maximum selector block. The tracking input ensures the availability of the controller during the period in which it is blocked by the minimum or maximum selector block. During this time, its integrator (and, hence, its output) is maintained at the value of the signal received, so that if the limit conditions are violated, it is able to respond immediately. [Track] is also used in conjunction with special actuators with positioning feedback.

Direct/reverse-acting control action [Actg]

[Actg] is a characteristic parameter of the controller and indicates the relationship between the setpoint deviation and the change in energy flow. A distinction is made between direct action and indirect [Actg].

- Direct control [Actg]: As the controlled variable rises, the controller output increases, and as the controlled variable falls, so the controller output decreases.
Example: Cooling or dehumidification – as the measured value rises above the setpoint, so the flow of energy is required to increase.
- Indirect control [Actg]: As the controlled variable decreases, the controller output decreases.
Example: Heating or humidification – as the measured value falls below the setpoint, so the flow of energy is required to increase.



Inversion [Inv]

[Inv] of the output signal is required, e.g., for air dampers. The outside air and exhaust air damper must close in response to an increasing heating demand. The inversion of the manipulated variable affects only the output signal [Yctr] and not the action of the controller.

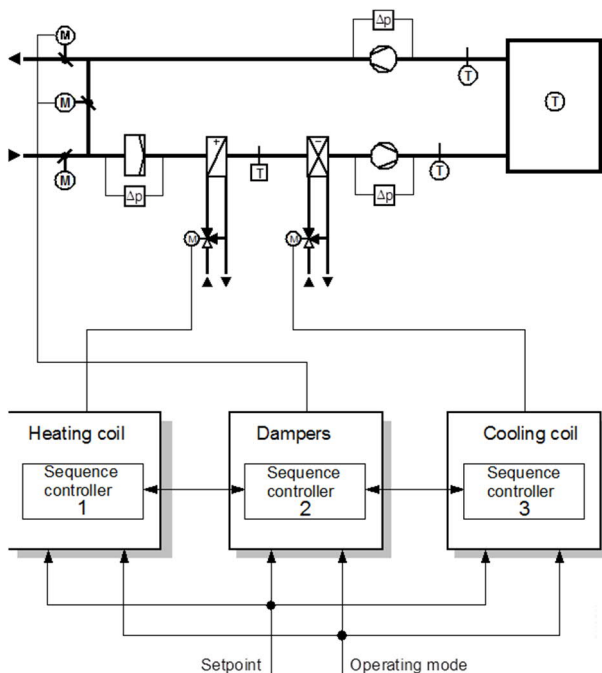
Sequence controller

Sequence controllers are used primarily in ventilation and air conditioning systems to control the temperature and humidity. Other applications are also possible, e.g., in heating systems.

Each controlled aggregate functional unit incorporates a universal PID controller function block, PID_CTR, as a sequence-controller element.

The statements made about the universal PID controller also apply to the use of the PID_CTR function block as a sequence-controller element.

The sequence-controller elements coordinate their own interaction independently. Interaction is coordinated with coordination signals [FmHigher] and [ToLower], which are mutually exchanged by adjacent sequence-controller elements. This is the only link between the sequence-controller elements. This process allows the setting of individual parameters for each individual controller or aggregate, and hence effective optimization of the entire plant.



Properties and design of sequences and sequence controllers:

- Each sequence may include any number of elements
- The setpoint for each element of a sequence can be defined separately, but set points must not be allowed to decrease in the direction from the heating sequence to the cooling sequence.
- The setpoint for energy recovery can be selected and is either at the midpoint between the setpoint of the first heating element and that of the first cooling element, or (depending on the method of energy recovery currently possible), it may be equivalent to the setpoint of the first heating element (if the extract air temperature is higher than the outside air temperature) or equivalent to the setpoint of the first cooling element (if the extract air temperature is lower than the outside air temperature).
- The gain of each sequence element can be influenced individually. In this way, e.g., the amplification factor (gain) of the energy-recovery element varies as a function of the difference between the extract air temperature and the outside air temperature, in order to achieve an almost constant loop gain.
- For each element, P, PI, PID, PD or on/off control can be selected. The control parameters for each element (controller gain, integral action time and derivative action time) can be adjusted individually.
- If all the sequence elements have the same parameter values, the sequence responds in exactly the same way as a single PI(D) controller whose output variable is distributed to individual aggregates within the plant.
- The controller output and the integrator of the sequence element is limited in the range [YctrMin] to [YctrMax]. For this purpose, the high limit of the last enabled sequence element of the heating and cooling sequence is limited with an anti-windup strategy (limitation of I/portion on manipulated variable limits). All other limit values are controlled by straightforward selection of the minimum or maximum value.
- The rate of change of the output of each sequence element is limited to the speed of the connected actuator. This helps improve control quality.

- The type of operation of each element (heating/cooling or humidification/dehumidification) can be selected individually for each element.
- Only one element of the sequence can have a controlling function. When the output of a controlling sequence element reaches [YctrMin] or [YctrMax], control is transferred to the nearest adjacent active element ("ON").

Naming convention

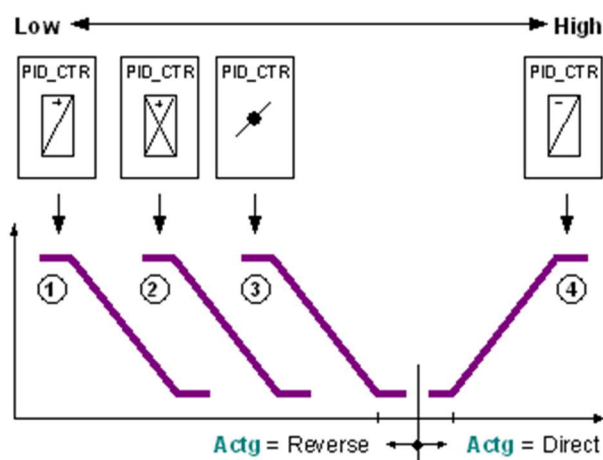
The term higher is applied to sequence elements that correspond to higher set points in the sequence diagram (normally cooling or dehumidification).

The term lower is applied to sequence elements that correspond to lower set points in the sequence diagram (normally heating, energy recovery or humidification).

Configuration of a sequence controller

Essentially, the sequence controller consists of individual PID_CTR blocks. with each PID_CTR block acting as a sequence-controller element for an aggregate.

The PID_CTR blocks are connected (from "Low" to "High") in the same order as the control sequences (1...n) of the sequence controller. Accordingly, the connection of the PID_CTR blocks must take account of the intended operating range (e.g., for heating) and the order of switching.



For example, aggregates:

1 = Re-heater, 2 = Pre-heater, 3 = Dampers, 4 = Cooling coil

Control series for heating: 3 ---> 2 ---> 1

Cooling control sequence: 4 ---> ...

The lowest sequence-controller element (Low) corresponds to control sequence 1, and the highest (High) to control sequence n.

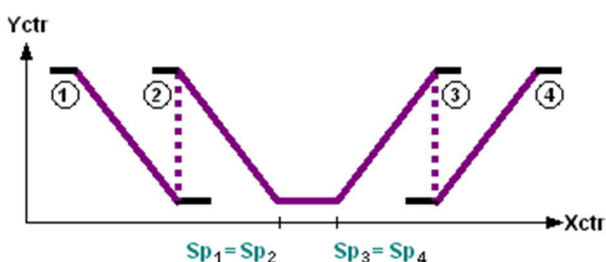
The lowest sequence-controller element controls a reverse-acting aggregate (if used).

The type of operation may also be reversed during normal operation, (e.g., for energy recovery) but the order of the sequences must not be affected.

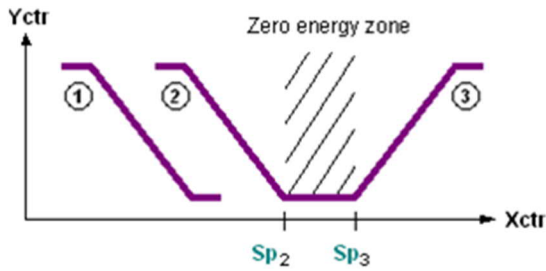
In the sequence controller, the set points [Sp] of sequence-controller elements (1...n) must increase incrementally:

$$[Sp]_1 \leq [Sp]_2 \leq [Sp]_3 \leq \dots \leq [Sp]_n$$

In the transition from one control sequence to the next, continuous control is maintained if all the control sequences with the same type of operation (direct or reverse acting) also have the same setpoint.



When the type of operation changes, the neutral zone is defined by the set points (e.g., heating setpoint / cooling setpoint).



Options for connecting sequence controller elements

The PID_CTR blocks can be connected to form a sequence controller via:

- Direct connection
- SEQLINK connection

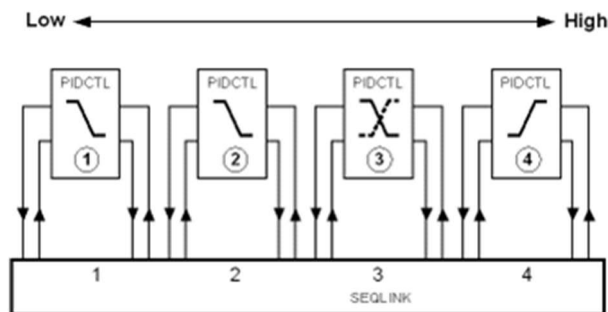
Direct connection

The individual PID_CTR blocks are connected directly with each other. The [ToLower] pins are connected to the [FmHigher] pins, and the [FmLower] pins are connected to the [ToHigher] pins.



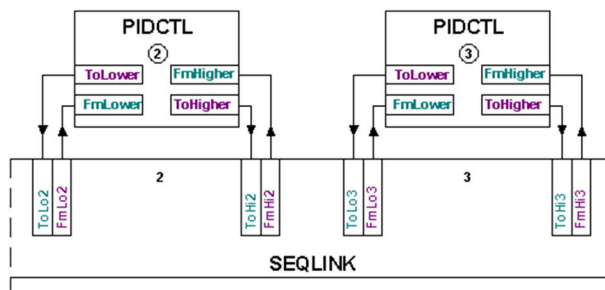
SEQLINK connection

With this method, the individual PID_CTL blocks are connected via the SEQLINK block. The sequence linker block SEQLINK is a wiring block with no function other than that of connecting other blocks.



The connection is made between the pins of block PID_CTL and a location on the SEQLINK block. The order in which the PID_CTL blocks are connected must be the same as that of the sequence. The connections to the SEQLINK block need not be continuous: connected pins and unused pins may be interspersed.

For example, 1 = Re-heater, 2 = Pre-heater, 3 = Dampers, 6 = Cooling coil.

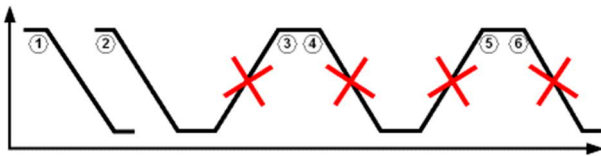


This method of connection is used to interconnect PID_CTR blocks on different charts, or in cases where individual project-specific sequence-controller elements or aggregates are to be deleted from an off-the-shelf solution (CAS library).

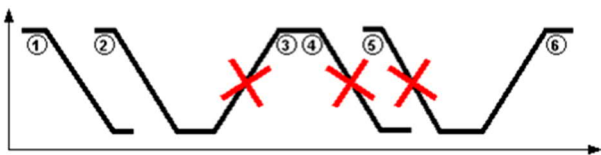
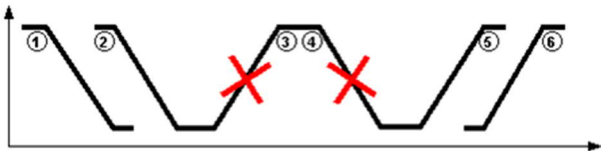
Communication between one sequence controller element and another flows via the pins [ToLower] → [FmHigher] and [ToHigher] → [FmLower].

The block recognizes configuration errors and shows these at the Token State output [TknSta]. If, e.g., the control action [Actg] of an individual sequence-controller element is incorrectly set, the associated sequence controller element is disabled and an error message is displayed.

Example: Output from elements 4 and 6 [TknSta] = HEL_CSEQ Output from elements 3 and 5 [TknSta] = CEL_HSEQ:



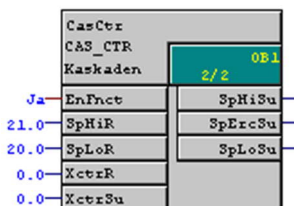
Examples of automatically deactivated sequence elements:



In all the examples illustrated, several aggregates are deactivated. This is a precaution, as the sequence elements cannot determine which of the aggregates has incorrectly set parameters. For this reason, the aggregates are disabled one after the other until there is a clear transition to the next sequence.

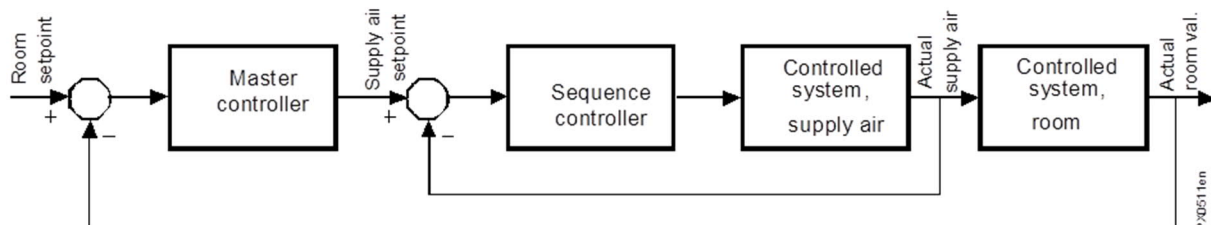
Cascade control

The CAS_CTR block integrated into the Desigo system is a PI master controller for room supply air cascade control. It delivers three supply air set points on the basis of the difference between the measured room temperature and the room setpoint.



The following functions are integrated into the block:

- Facility to select P or PI control
- Gain and integral action time (can be configured)
- Low supply air setpoint for the reverse-acting part of the sequence
- High supply air setpoint for the direct-acting part of the sequence
- Supply air setpoint for energy recovery
- Min/Max setpoint limit control (supply air setpoint)
- Selection of type of operation for heat recovery
- Initial value for the integrator can be defined



Compared with control without a cascade, e.g., cascade control improves the dynamics of the control process. If the temperature in a ventilated room is below the setpoint, e.g., the supply air temperature must be increased, at least for a brief period, in order to raise the temperature to the room setpoint. This can be achieved by measuring and controlling not only the room temperature, (that is, the value which actually concerns the user), but also the supply air temperature, whose setpoint depends on the difference between the room setpoint and the room temperature.

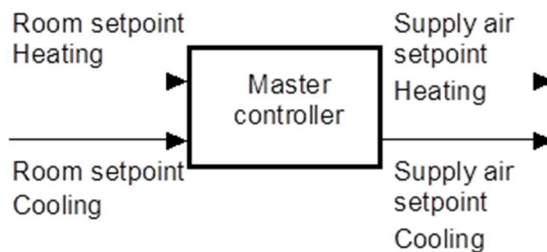
If the room temperature is lower than the room setpoint, the supply air setpoint is adjusted in proportion to the room control differential, and the supply air temperature is increased via the supply air control loop.

The master controller generates the setpoint for the auxiliary variable (e.g., the supply air temperature) on the basis of the difference between the primary setpoint and the primary controlled variable (e.g., the room setpoint and the room temperature).

The master controller must include an integrator function (I component), because even under static conditions (that is, when the measured value and the setpoint are equal) there is generally a negligible control deviation, which means that the controller output must be at a different operating point. For improved control dynamics, a P-component should be connected in parallel with the integrator. This is why the master controller in this case has a PI control structure.

Even when the primary controlled variable (room temperature) is identical to its setpoint, the auxiliary controlled variable (supply air temperature) must generally be at a value other than 0, (that is, setpoint \neq 0). This is only possible if the output of the master controller is not equal to 0, even if the P component = 0. In other words, the master controller must have an I-component which remains constant when the control differential = 0. This is why the master controller has a proportional and an integral component. It is a numerical PI controller for use as a master controller in a room/supply air cascade.

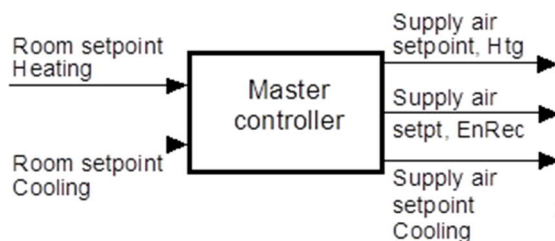
To save energy in the ventilation plant, various room set points are selected for different types of air handling (heating/cooling and humidification/dehumidification). The master controller in the cascade must therefore be able to generate different supply air set points, depending on how the kind of air treatment (heating/cooling or humidification/dehumidification).



The supply air controller must determine whether the heating or cooling sequence is to be activated and the decision-making strategy does not affect the calculation of the two supply air set points. Within the cascade control loop, the supply air set points always move parallel to each other, and their offset is determined by the integral component.

If the air handling plant includes an energy recovery aggregate, this aggregate may be either reverse-acting (e.g., heating) or direct-acting (e.g., cooling) depending on the relationship between the condition of the outside air and the condition of the exhaust air.

To avoid external calculation of the energy recovery setpoint, this, too, is done by the master cascade controller, and made available to the energy recovery aggregate, if there is one, at a separate output pin:



In a humidity control system with various physical control variables, the initial value of the integrator should be predefined.

Example:

If the humidity of the supply air is measured with an absolute humidity value [g/kg], while the room air humidity is measured in terms of relative humidity [%Hu], an initial value must be defined for the I-component, otherwise the mean value from [SpLoR] and [SpHiR] will be used as the initial value. If the room set points are expressed in terms of relative humidity, then the initial value for the integrator will start at a numerically high value, and decrease as a function of the preset integral action time [Tn]. The result of this can be that even if the room needs to be dehumidified, the humidifier is enabled in the controller start-up phase until the integrator reaches its correct value.

To prevent this, the current measured supply air humidity value is linked to the initial value of the integrator, or a fixed parameter value is defined for the integrator.

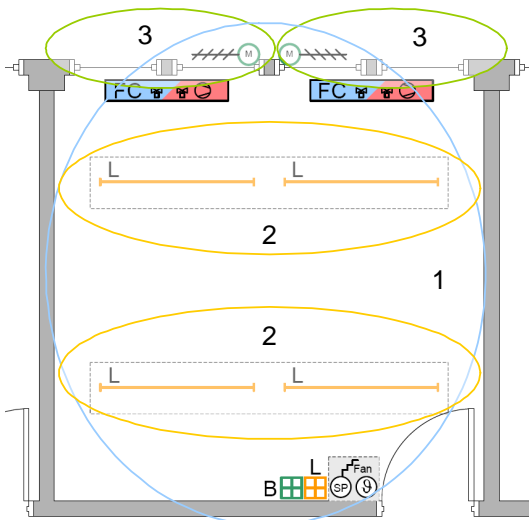
If control accuracy is critical (e.g., no deadband or zero-energy control zone), then the current measured value is linked to the initial value of the integrator, or a fixed parameter value is defined for the integrator.

5.5 Desigo room automation

Multiple mechanical and electrical installations (referred to as technical installations in this chapter) come together in one room. These typically are HVAC, lighting, and blinds. Each technical installation is automated and operated optimally from its perspective. For Desigo room automation, coordination of the individual technical installations must be optimized while considering that the same type of installation may exist several times in one room.

Room featuring:

1. HVAC zone (blue)
2. Lighting zones (yellow)
3. Shading/blinds zones (green)



HVAC zone

The room typically is considered 1 HVAC zone influenced via a common automation and control strategy regardless of number and type of installed HVAC plant components (e.g., radiator, chilled ceiling, fan coil unit).

Lighting zone

All lamps operated/automated together are grouped into a lighting zone regardless of number and type of the installed lamps. A room typically has one or several lighting zones.

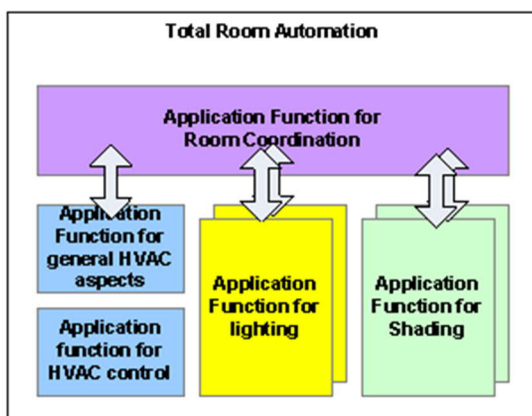
Shading zone

All shading products (blinds) operated/automated together are grouped into a shading zone regardless of number and type of the installed shading products. A room typically has one or several shading zones.

Desigo room automation and room coordination

Application function structure

Specific functionality is set up for each zone of each technical installation: The application functions. For Desigo room automation, this is supplemented by a room-wide function coordination called room coordination.



Room coordination basically has two application functions:

- Cross-technical installation coordination to ensure smooth functional interplay of the various installations
- Centralized, room-wide access point to operate and monitor a room

Cross-technical installation coordination

The application functions of the individual technical installations contain functionality required for technical installation-specific control. Additional functionality assuming coordination with other technical installations is part of room coordination. As a result, project-specific Desigo room automation requests and changes can be carried out without changes to technical installation-specific application functions.

Examples for coordination functions are coordination of HVAC and shading functions and coordination of shading and lighting functions.

Centralized, room-wide access point

Room coordination offers a centralized, room-wide access point to operate and monitor a room. This allows users to enter common data for several technical installations only once and retrieve them as a group.

Examples:

- Predefinition of the room operating mode (across all technical installations)
- Predefinition of a scene for the entire room
- Queries for general occupancy
- Common alarm for system alarms

The room coordination default solution influences the following functions:

Room operating mode

Various sources influence and determine the room operating mode:

- Centralized commands from scheduler programs or manual intervention
- Local commands from presence detectors or scheduler program override

Room coordination offers a centralized, room-wide access point to operate and monitor a room operating mode. The individual technical installations separately acquire all relevant information.

Scene

Scenes are defined to command several or all technical installations in a room via one single command, e.g., brightness of a lighting zone, or blinds positions in each shading zone can be defined for each scene. Room coordination:

- Controls a scene as per the predefined values
- Changes the predefined values

Both are carried out by the room user.

Thermal room load analysis

Room coordination supports room temperature control via blinds control. The various HVAC data is analyzed to determine the thermal room load and the associated signal definition for blinds control:

- Load if energy must be supplied to the room via the blinds position
- Unload if no additional energy must be supplied to the room via the blinds position

Blinds control determines the optimal blinds position in dependence of room occupancy and solar position (thermal radiation and glare).

Green Leaf (RoomOptiControl)

Manual room user manipulations (e.g., manual lighting and shading commands, or manual changes to the room temperature setpoint) can result in inefficient energy operation. Each zone and each technical installation is checked for inefficient definitions to be pointed out to the room user. Room coordination then summarizes the results and visualizes them on the room operator unit. The room user can then reset all manual entries (which lead to inefficient plant operation) by one single pressure of a button.

Room common alarm

One common alarm is set up for each room to keep down the number of set up system alarms. To this end, room coordination acquires status information (normal/alarm) for each zone and each technical installation, and determines the room-wide alarm state as a common alarm.

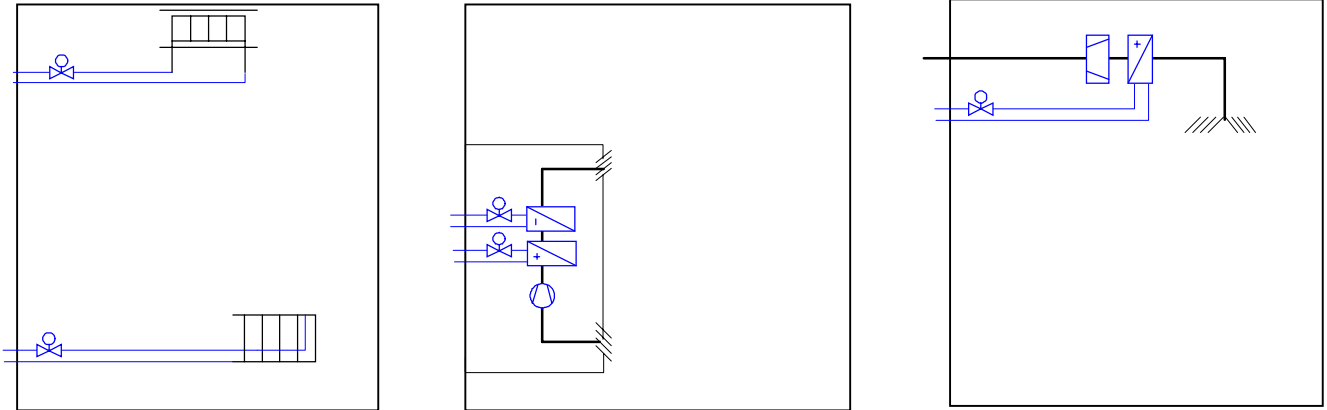
HVAC room control

HVAC plants and their HVAC devices in the room influence the climate in closed rooms.

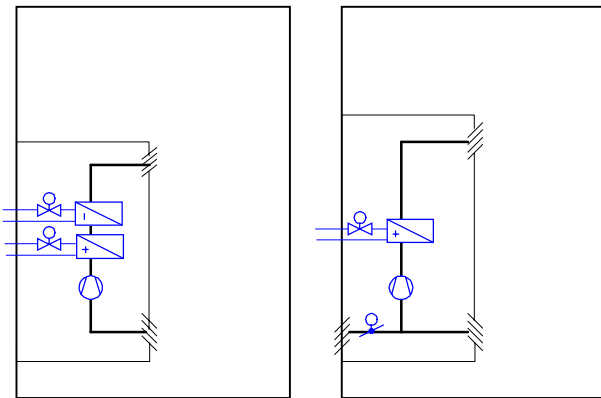
HVAC plants in rooms are used to:

- Maintain a temperature range suitable for building occupancy
- Control other control variables such as humidity and air quality
- Efficiently operate HVAC plants in the room

HVAC plants in the room are grouped into plant families, radiators (right), Fan coil units (center), VAV (left), differentiated by mechanical design and functioning:



The members of the related HVAC family differ only marginally:



HVAC supply chain requirements

HVAC plants in a room consume energy. The supply chains outside the room supply air, water, or electricity to the room. Linked existing energy sources and consumers are called supply chain. An air supply chain or a water supply chain thus is an HVAC system with a supply/consumer relationship to the HVAC plant in the room.

The supply equipment typically supplies more than one room, and the HVAC plant in the room often is a consumer of multiple supply chains.

HVAC control basically has the same objectives as the entire HVAC plant:

- Maintain the room temperature in the selected comfort range
- Adapt the room temperature range to room user needs
- Supply, extract, and recirculated air to satisfy air quality and comfort needs
- Adapt the air flows to room user needs

Energy saving requirements:

- Devices for sequential control of a heating and cooling sequence and thus:
 - Preventing sequence overlap (simultaneous heating and cooling)
 - Using the most efficient energy source
- Reducing the temperature as soon as comfort mode no longer is needed
- Reducing ventilation as soon as it is no longer needed

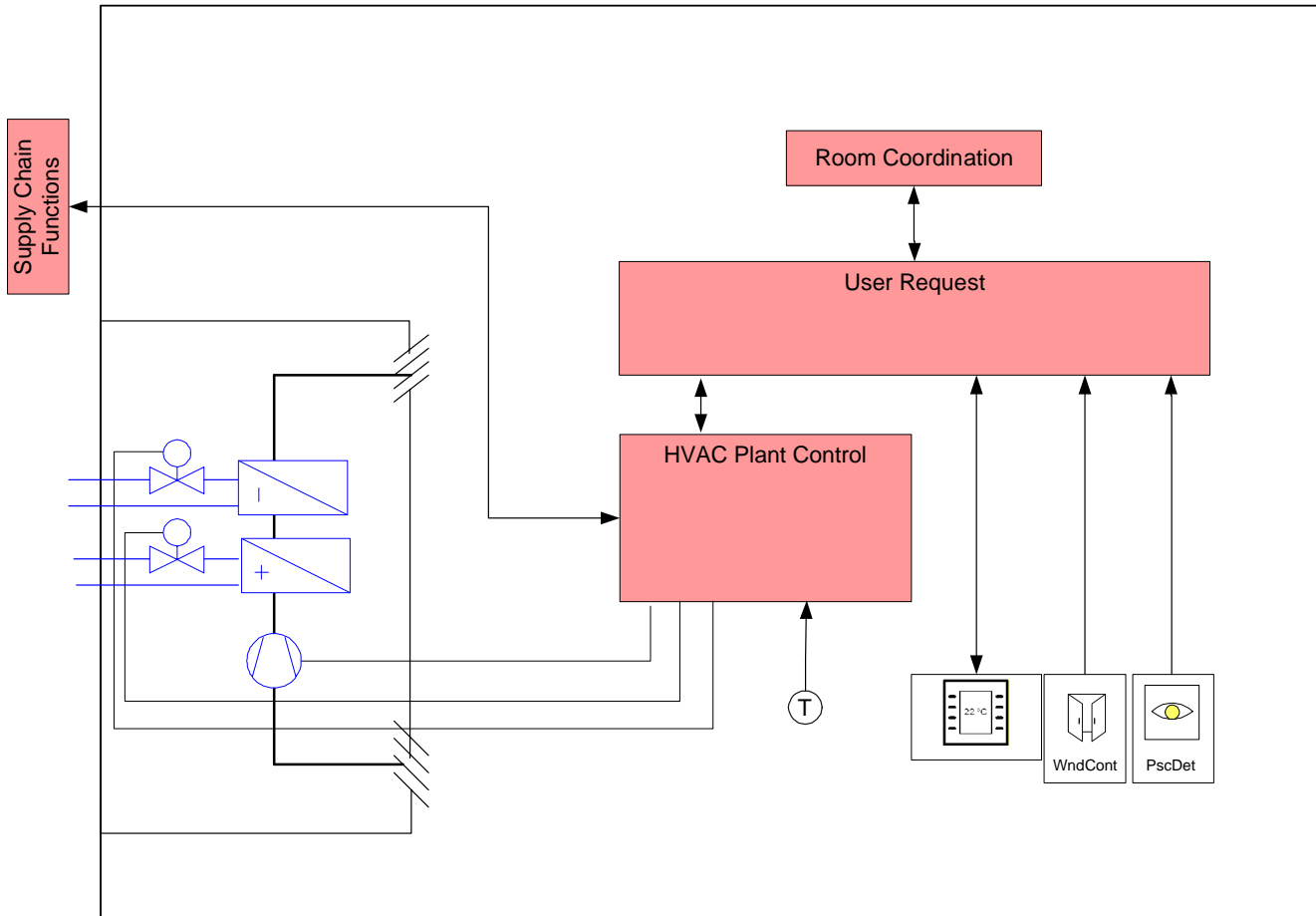
Coordination of the HVAC supply chain:

- Operation of supply chain equipment as per user demand
- Optimization of operating levels (temperature, pressure) of the supply plant
- Prevention of damages to HVAC equipment

HVAC control structure

An HVAC control application in the room is connected to the following:

- HVAC plant in the room via sensors and actuators
- Room coordination application
- Centralized coordination application for HVAC supply chain(s)
- Building operator via BAC workstations
- Building automation and control functions for scheduling
- Room user



The HVAC control application in the room consists of two parts:

- Application function for user requirements
- Application function for HVAC plant control

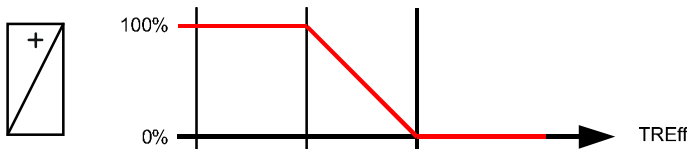
The HVAC plant control contains a control module (CFC) that implements the control functions associated with the HVAC device.

Control concepts

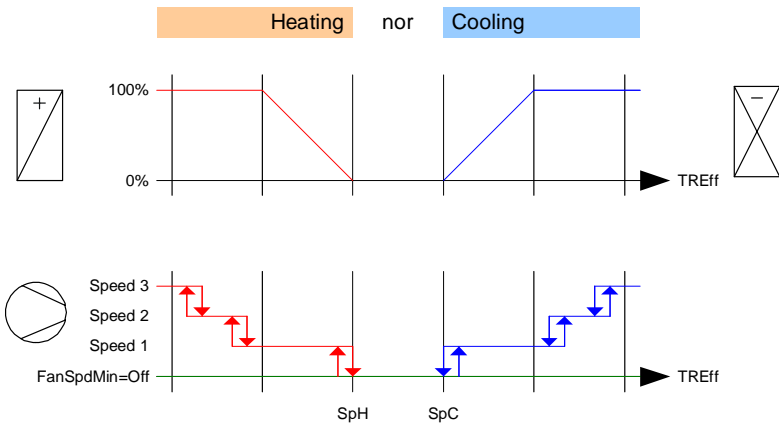
The physical room conditions are controlled by a combination of control methods (setpoints by operating mode).

Sequence control

Algorithms for room temperature sequence control operate the heating and cooling equipment within applicable limits. The algorithm for one single heating element is as follows (e.g., radiator):



Below is an illustration of the temperature control sequence for a more complex HVAC plant in the room. The charts show the segregation of heating and cooling control sequences and associated setpoints and sequencing of heat convection by fan air flow or associated switching stages.



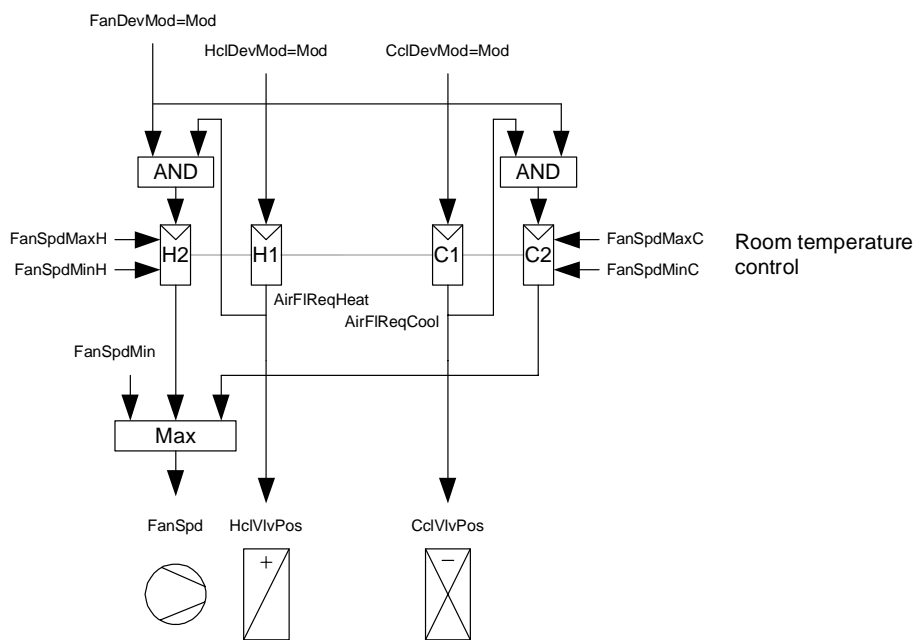
Individual temperature sequence controllers are assigned to each heating and cooling element. They intercommunicate to achieve required sequencing.

Open-loop control

Additional interactions between HVAC devices implemented via open-loop control functions are required in an HVAC plant in the room. The open-loop control functions feature two basic interactions:

- Support: Heating coil and cooling coil require the fan to run on the stage/speed required for their operation.
- Lock: The electric heating coil is locked to ensure that it cannot be operated without air flow.

Open-loop control and sequence controller are used together to implement the above, typical control sequence. The following image shows the connection between controller and actuating devices (this does not correspond to the actual program structure).



Operating modes

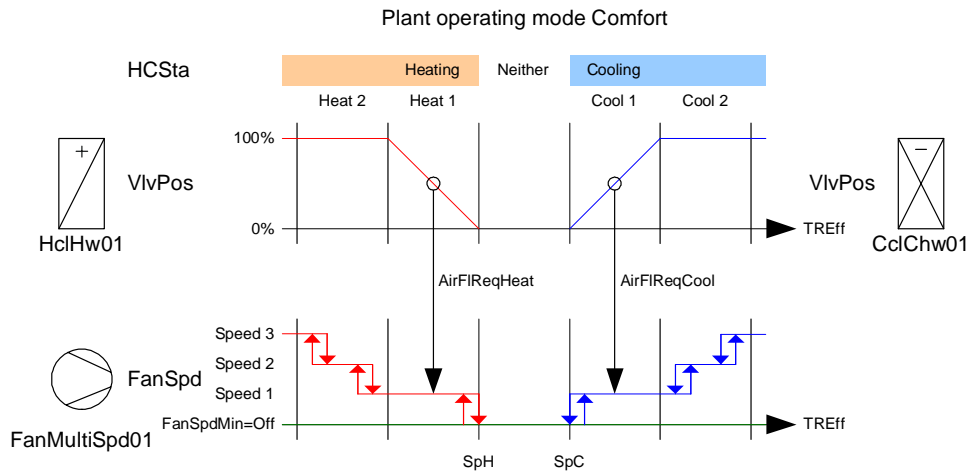
The HVAC plants in the room adapt to the room's comfort requirements, e.g., ventilation is:

Active while the room is occupied

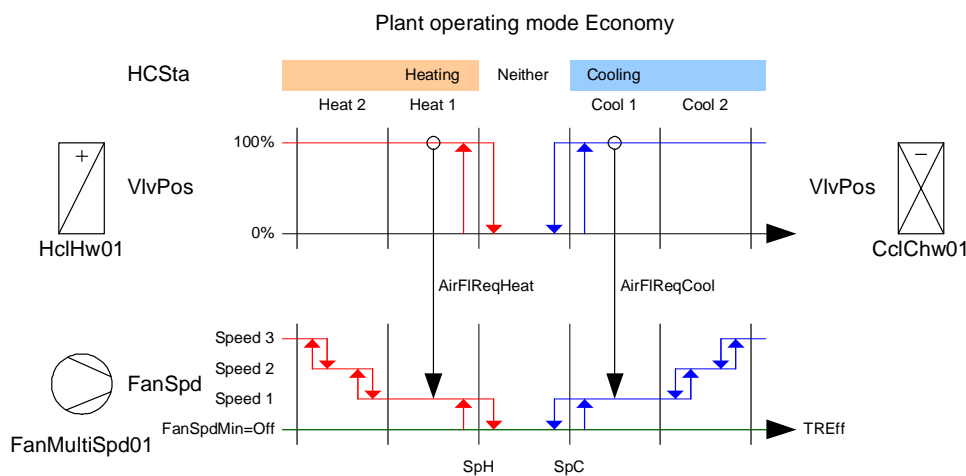
Off, as soon as the room no longer is occupied

The following illustrations show sequence control for an HVAC plant in the room for operating modes Comfort and Economy. Sequence control acts on heating and cooling equipment and a multi-speed fan.

Control sequences in the Comfort operating mode:



Control sequences in the Economy operating mode:



The available operating modes determine both operation and basic control strategy in the automation and control system at three different levels:

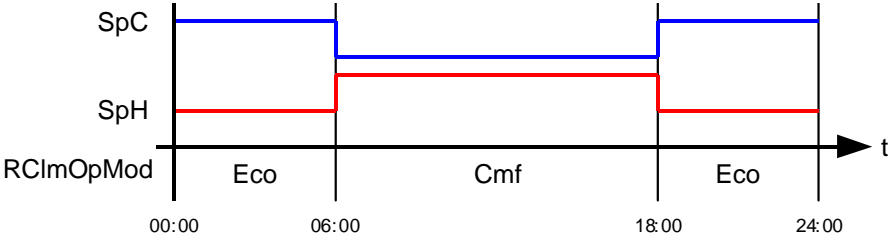
- The room operating modes determine the operation of HVAC equipment in a room in terms of current use by the user. The room operating modes defined for the room are available in all HVAC control applications in the room.
- The HVAC plant operating modes determine the operation the HVAC plant in the room with regard to existing, physical plant processes. The HVAC plant operating modes are defined specifically for 1 HVAC plant in the room.
- The device operating modes determine the operation of the HVAC devices in a room by predefining their tasks and implementation method. The device operating modes are defined specifically for each individual HVAC device.

Plant and device operating modes of a plant with heating coil, cooling coil, and fan

Project-specific adaptations of both plant and device operating modes can be implemented by adapting the operating mode table.

| Plant operating mode | Fan operating mode | Heating coil operating mode | Cooling coil operating mode |
|----------------------|--------------------|-----------------------------|-----------------------------|
| Off | Off | Off | Off |
| Comfort | Modulating | Modulating | Modulating |
| PreComfort | Modulating | Modulating | Modulating |
| Economy | Modulating | Two-position | Two-position |
| Protection | Modulating | Two-position | Off |
| Heat up | Modulating | Two-position | Off |
| Cool down | Modulating | Off | Two-position |

In addition, setpoints and setpoint limits define room and device operating modes. They can vary depending on the selected HVAC plant operating mode. Four different setpoints are provided for heating and cooling in the room.

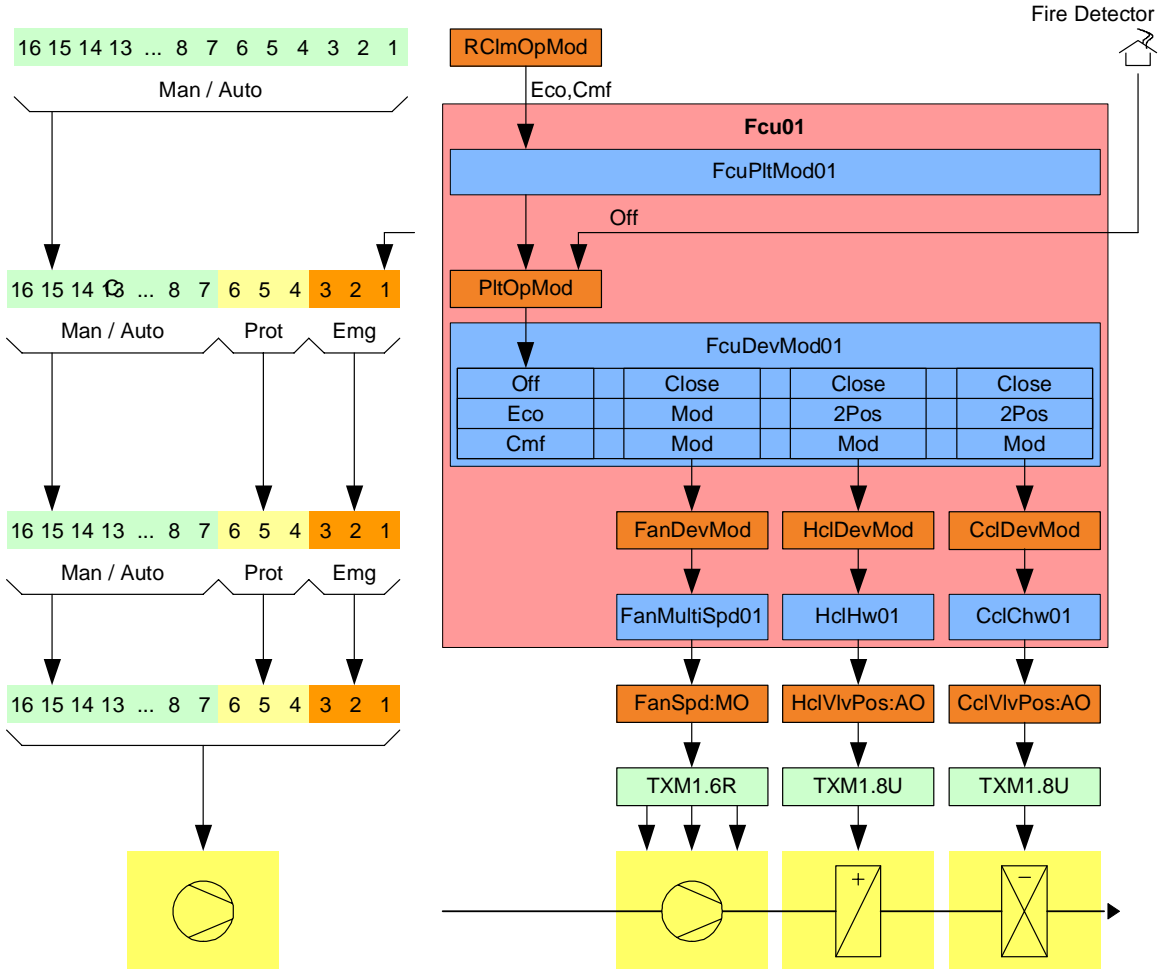


The HVAC control applications in the room dynamically enable and disable the setpoints to achieve the desired combination of energy-saving Economy and demand-based Comfort operating mode.

Command priorities

An HVAC control application simultaneously achieves several goals. Functions with different objects may conflict when they are implemented. In this case, the command priority determines which command value has priority in the priority array of the BACnet objects.

HVAC control applications in a room are programmed to accept commands at many different levels, including operating mode variable level. As a result, HVAC control applications control the controlled output objects at a priority commensurate with the active priority of the operating mode variable. The following figure shows how commands and priorities are passed in the application.

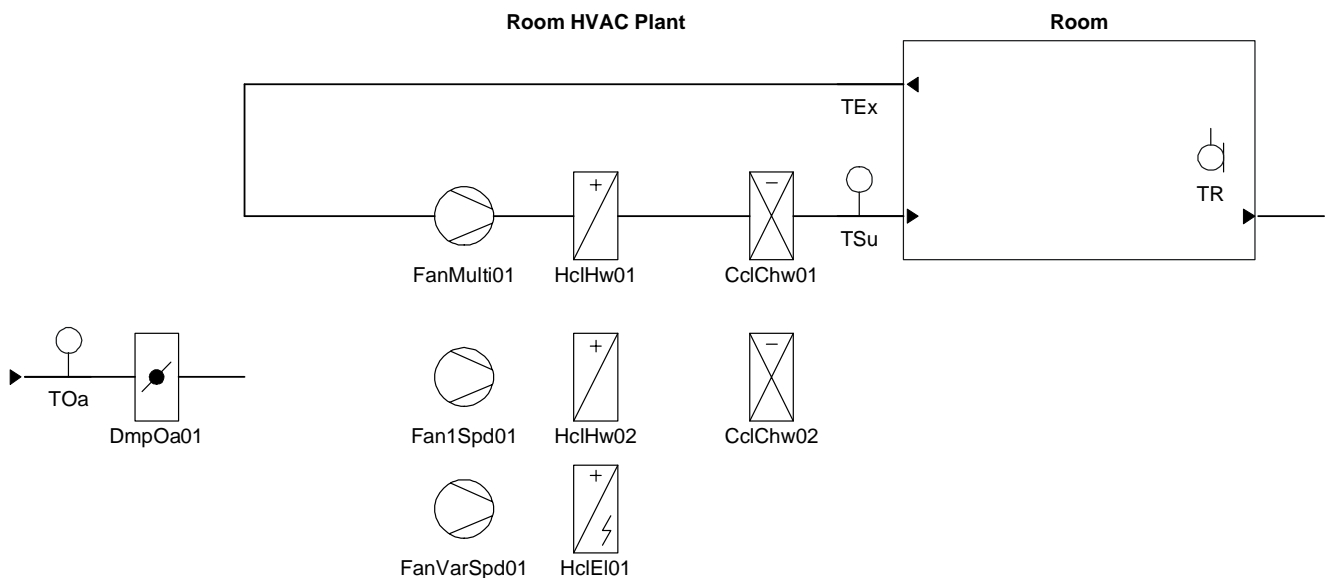


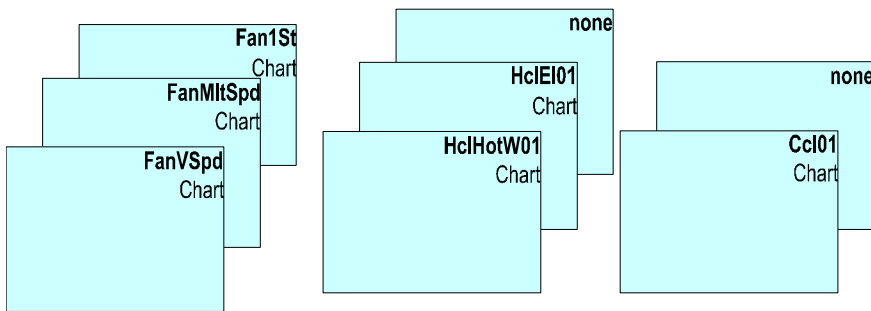
The BACnet objects in the system support 16 priority levels. The HVAC control applications apply these levels as follows:

| Priority | Purpose assigned to the level | Use in HVAC library |
|----------------------|---|--|
| Emergency mode 1 | Manual commands related personal safety | None |
| Emergency mode 2 | Automatic commands related to personal safety | Propagated response to Emergency mode commands |
| Emergency mode 3 | Unassigned - additional level for commands related to personal safety | None |
| Protection mode 4 | Manual commands to avoid damage to equipment | None |
| Protection mode 5 | Automatic commands to avoid damage to equipment | Programmed response to equipment safety conditions |
| Minimum On/Off 6 | Commands to avoid damage by short cycling equipment | None |
| Manual operating 7 | Manual commands through switches on equipment | None |
| Manual operating 8 | Manual commands through BAS workstation | None |
| Automatic control 9 | Unassigned - commands for comfort and energy conservation | None |
| Automatic control 10 | Unassigned - commands for comfort and energy conservation | None |
| Automatic control 11 | Unassigned - commands for comfort and energy conservation | None |
| Automatic control 12 | Unassigned - commands for comfort and energy conservation | None |
| Manual operating 13 | Manual commands through room unit | Programmed response to inputs from occupants |
| Automatic control 14 | Unassigned - commands for comfort and energy conservation | None |
| Automatic control 15 | Typical automatic commands for comfort and energy conservation | Typical automatic commands |
| Automatic control 16 | Unassigned - commands for comfort and energy conservation | None |

Adaptation to another HVAC plant

An HVAC control application comprises several different members of an HVAC room family. It contains application-specific components (CFC) matching existing HVAC devices in the room. Components no longer matching existing HVAC devices in the room are either added, removed, or replaced to control a slightly different HVAC plant with different HVAC devices.





Often, more must be done than merely adding or removing components (CFCs). If, an HVAC device, e.g., is to be added, the following must be added or removed:

- Information in the operating mode table
- Corresponding BACnet objects to operate the new device

Shading control

Products and requirements

Suitable façade products and intelligent control allow for optimum satisfaction of various requirements for shading.

Façade products and their control required to protect against environmental influences or to make use of the same are the primary issue:

- Shading to protect against glare
- Using daylight
- Using solar energy for heating
- Shading to protect against overheating
- Protection against rain

Other requirements may be:

- Intrusion protection
- Protection of privacy

Façade product control in addition must protect persons and equipment against the façade products themselves. Examples:

- Drive up blinds in case of fire to open an escape route
- Protect against collision (e.g., in the event of outward-opening doors)

Façade control protects the façade products and their functionality against environmental damages caused, e.g., by rain, wind, or frost.

The market knows many different façade products such as roller shutters, blinds, awnings, etc. to satisfy the various requirements. The different properties of the products are included in the respective control functions. The following figure shows a few typical façade products (from left to right):

- Horizontal blinds
- Roller shutter
- Vertical blinds
- Drop-arm awning
- Vertical awning
- Sliding-arm awning



Influences on blinds control

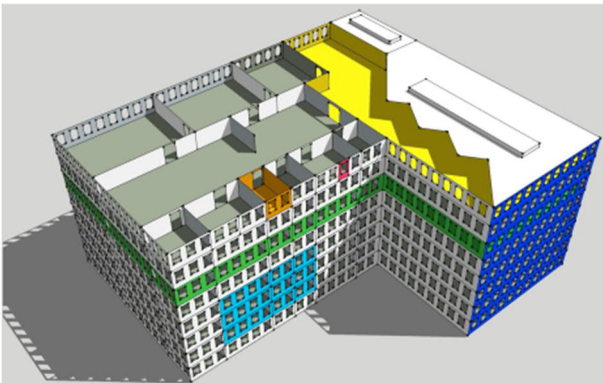
Blinds control requires much information on environmental influences and user interactions to be able to best satisfy requirements.

The blinds control can be influenced by, e.g.:

- Smoke, fire alarm
- Maintenance switch
- Wind, rain, humidity, temperature
- Intrusion alarm
- Date/Time
- Solar radiation
- Geographical position
- Horizon limitation
- Presence detector
- Local operator
- Saving and retrieving scenes
- Central operation (operation, scenes, override)
- Designo CC
- Commissioning/Test

Blinds position on a building, room purpose, and allocation of rooms to an organizational unit determine the type of information acting on blinds control. Example:

- Wind speed monitoring acts on all blinds of a building or building wing
- Automatic shading acts on all blinds of façade or part of a facade
- A scheduler program acts on all rooms of a renter
- Local manual operation acts on all blinds of a room, or on a single blind



Color key:

- Gray: Complete building
- Blue: Façade or part of a facade
- Green: Rooms of a renter, e.g., one floor
- Orange, red: Local, manual operation

The functions are grouped into local and central functions depending on whether the function acts on one or multiple blinds in a room, or on an entire group of blinds, e.g., on all blinds of a facade.

Grouping by local and central functions for the examples from the figure above

| | Local manual operation | Automatic shading | Wind speed monitoring | Scheduler program |
|------------------|--|--|---|---|
| Central function | n/a | Determination of optimal shading position in dependence of sun position | Measuring of wind speed Monitoring of wind speed Commanding of wind protection position | Commanding of a position in dependence of daytime |
| Local function | Commanding of manual position Positioning of blinds | Decision on which position is commanded automatically Positioning of blinds | Positioning of blinds | Positioning of blinds |

Control concept

The control concept is based on the following:

- Grouping into autonomous functions determining a set position for the blinds
- Priority assignment to individual functions
- Evaluation of all functions and decision in favor of specific blinds position based on priorities

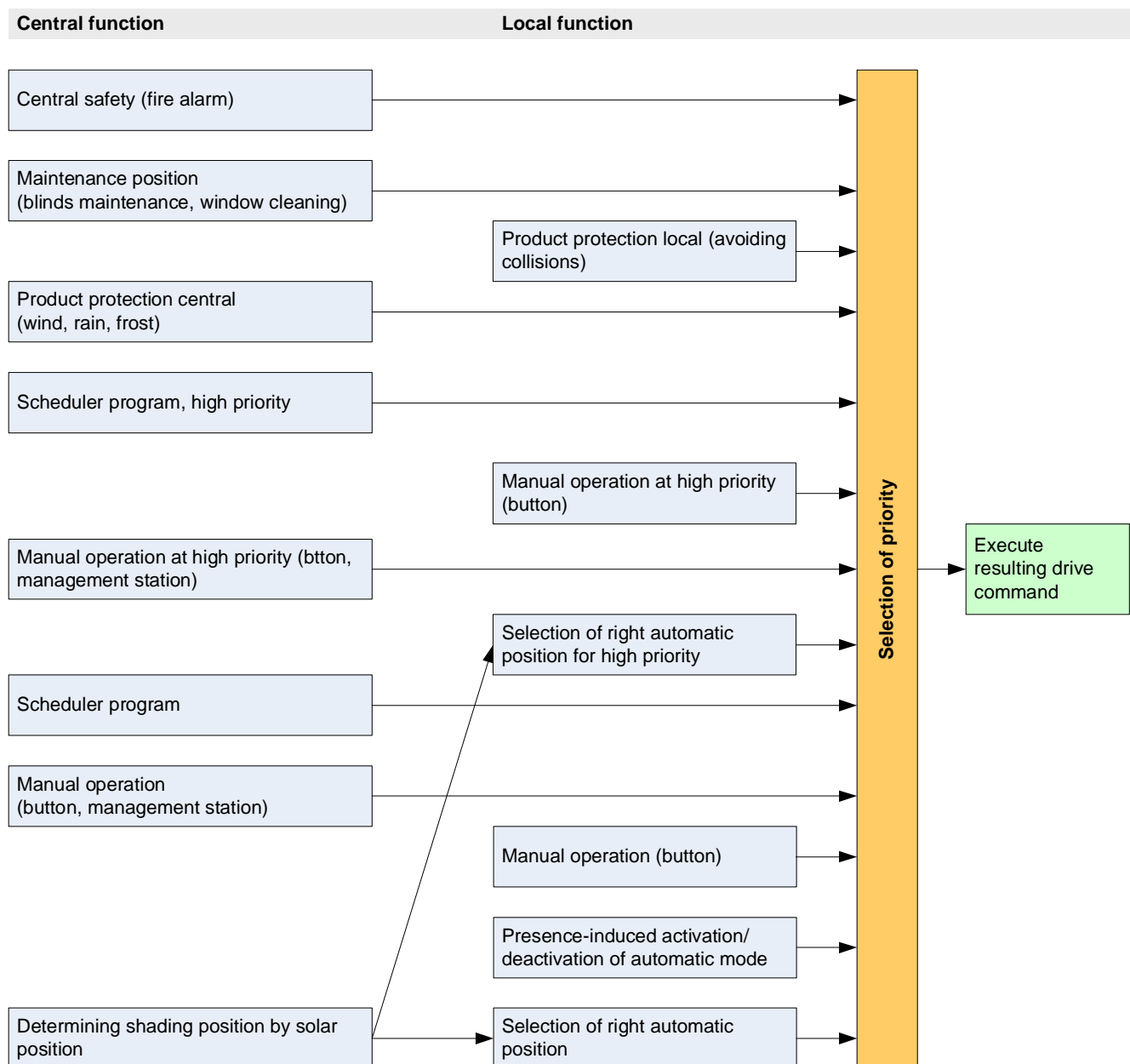
Overview of autonomous functions to control blinds

Priorities depend on plant requirements. The table shows the typical priorities in ascending format.

| Function | Description |
|---|--|
| Automatic shading | Automatic determination of optimum blinds position based on current room use, solar radiation, outdoor brightness, solar position, and HVAC status. In simple terms, this function prevents glare in occupied rooms and uses solar energy for heating in unoccupied rooms, or protects the building against undesired heat-up. |
| Manual operation (room, central) | Manual operation allows users to themselves determine the blinds position via buttons. If manual operation overrides a lower-priority function, a scheduler program or local presence information will reactivate the function. |
| Presence-based influence (room) | Locking of automatic operation upon entering a room, and activation of automatic operation upon leaving a room. The presence-based function generally acts on the same priority as manual operation. |
| Scheduler program | A scheduler program opens, closes blinds at specific times, or commands them to a specific shading position. Furthermore, automatic operation can be activated or deactivated via scheduler program. Another priority may need to be commanded depending on purpose. If, e.g., automatic operation should be activated at noon, manual operation must be overridden by allowing the scheduler program to act on the priority for manual operation. If the blinds are to be closed at night without allowing for manual operation, a higher priority must be commanded. |
| Automatic shading at high priority | For example, to prevent overheating it may be necessary to use automatic shading at higher priority, which limits or prevents manual operation in certain situations. |
| Manual operation at high priority (room, central) | Manual operation at high priority allows for positioning blinds and overriding low-priority functions. For example, local operation can be overridden during, e.g., a presentation. Or it can be ensured that neither automatic shading nor a scheduler will drive the blinds up or down at an undesired time. |

| Function | Description |
|-------------------------------|---|
| Product protection, local | Risks impacting a blind only, e.g., protection against collision with a service door opening outward, are included in local product protection. |
| Product protection, central | Environmental influences impacting a group of blinds are included in the central functions for product protection. A common function in this category is protection of blinds against damage from strong winds. |
| Maintenance position, central | For maintenance or cleaning, blinds are commanded or blocked to a specific position at high priority enabling staff to carry out all required work without risking injury due to moving blinds. |
| Protection, central | Blinds can be moved/driven up to provide escape or access routes for emergency personnel in the event of a fire. |

A very simple control contains just one or two functions; a complex plant may use many or all available functions. In addition, the response of individual functions may require parameterization depending on the requirements. The following figure shows an example of a plant including all functions.



Lighting control

Products and requirements

Suitable lighting products and intelligent control allow for optimum satisfaction of various requirements.

Lighting products and their control are the primary means to create optimum lighting conditions for building users:

- Optimum workspace conditions (bright or darkened rooms)
- Optimum lecturing/teaching conditions (presentation)
- Comfort in living spaces
- Mood lighting in entertainment spaces (restaurants, bars, etc.)

Other requirements may be:

- Energy savings
- Lighting of objects, products
- Façade lighting
- Intrusion protection

Lighting products control in addition must ensure the safety of persons. Examples:

- Switching on lights in case of fire
- Escape route lighting

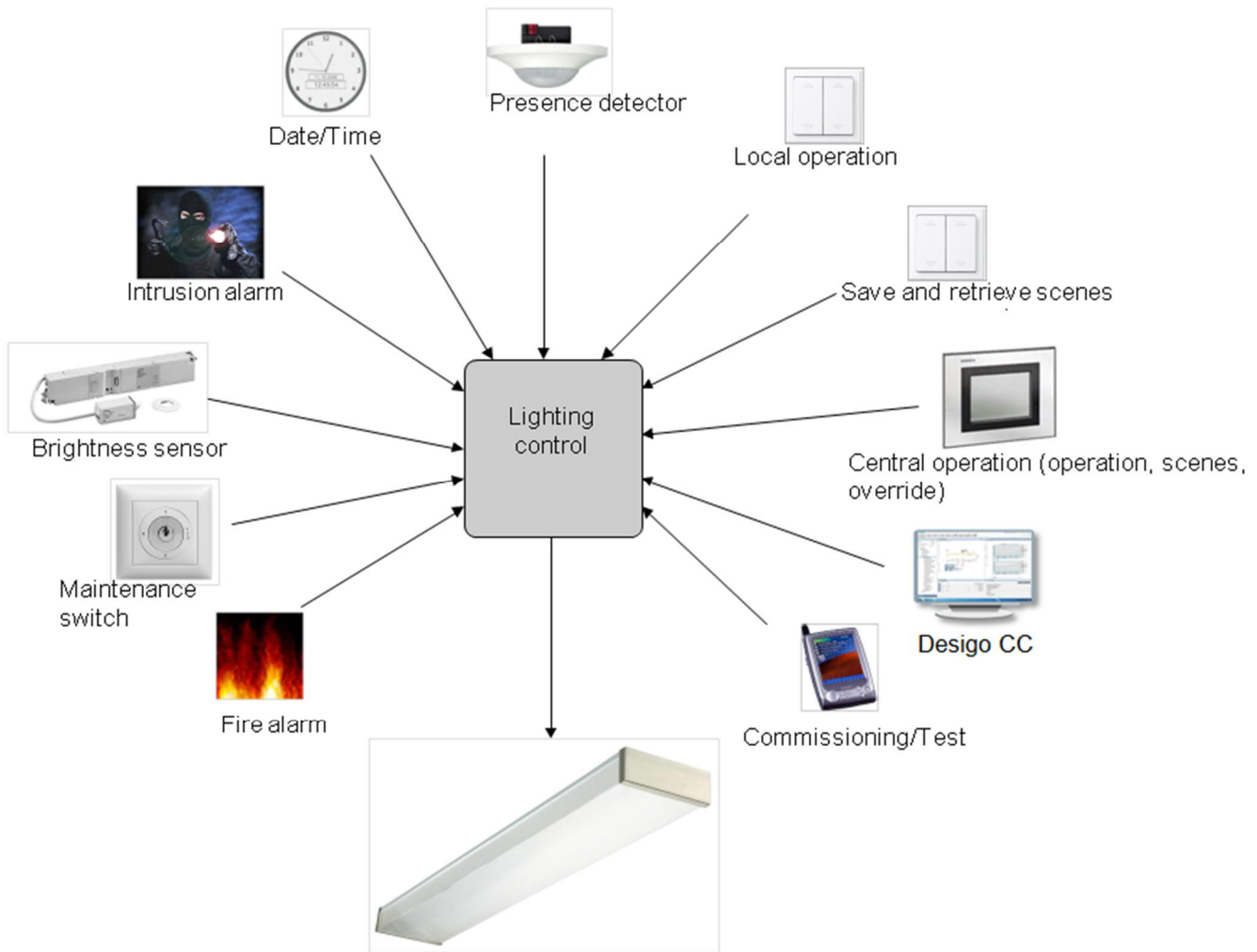
A multitude of different lamps exists to satisfy the various needs, such as:

- Incandescent light bulbs
- Halogen lamps
- Fluorescent tube lamps
- Compact fluorescent tube lamps
- Metal halide lamps
- LEDs

For comprehensive information on lighting products and their application, see the e-learning module *Lighting basics* (B_B01RA).

Influences on lighting control

Blinds control requires much information on external influences and user interactions to be able to best satisfy requirements. The following figure shows an overview of the influences that may be considered as part of lighting control.



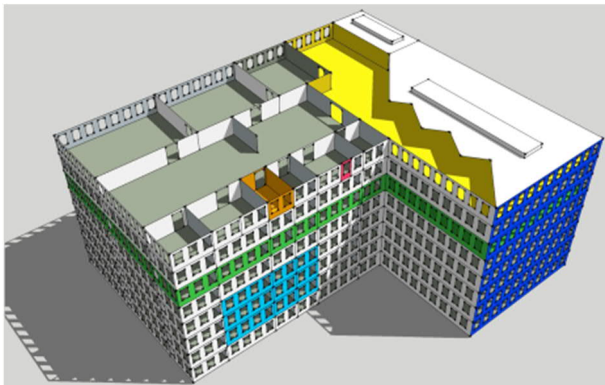
Lighting product positioning in a building, room purpose, and allocation of rooms to an organizational unit determine the type of information acting on lamp control. Example:

- A fire alarm acts on the entire building
- A scheduler program acts on all rooms of a renter
- Local manual operation acts on all lighting of a room, or on individual lamps

Gray: Complete building

Green/yellow: Rooms of a renter, e.g., one floor

Orange: Local, manual operation



The functions are grouped into local and central functions depending on whether the function acts on one or multiple lamps in a room, or on an entire group of lamps, e.g., on all lamps of a renter.

Grouping by local and central functions for the examples from the figure above

| | Local manual operation | Fir alarm | Scheduler program |
|------------------|--|--|--|
| Central function | n/a | Fire alarm reception Commanding of On-command | Commanding of On/Off command in dependence of time |
| Local function | Commanding of manual brightness Adapting lighting | Switching on lighting | Switch on/off lighting |

Control concept

The control concept is based on the following:

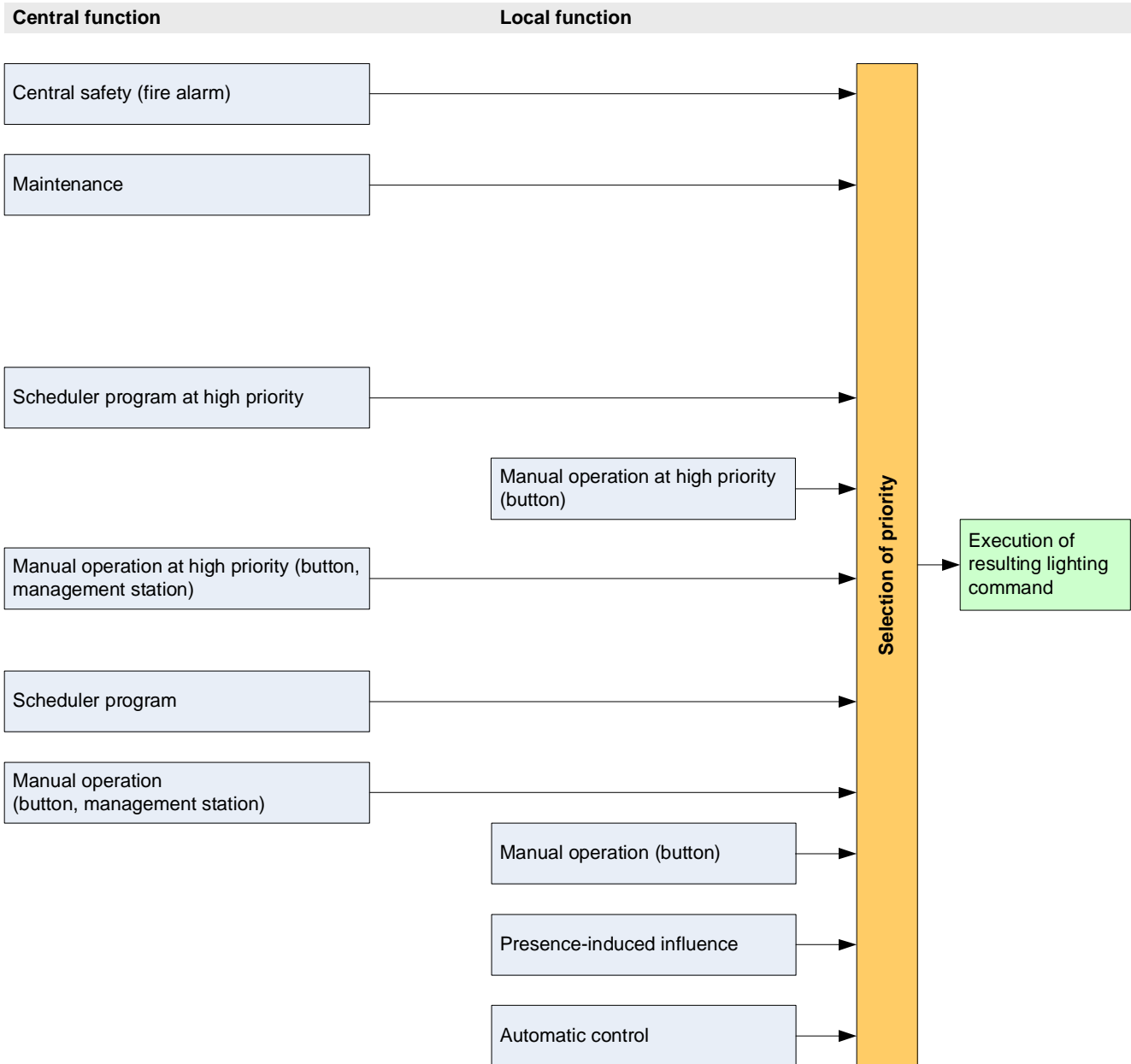
- Grouping into autonomous functions determining a command for lighting
- Priority assignment to individual functions
- Evaluation of all functions and decision on the state of lighting based on priorities

Autonomous functions to control lighting

Priorities depend on plant requirements. The table shows the typical priorities in ascending format.

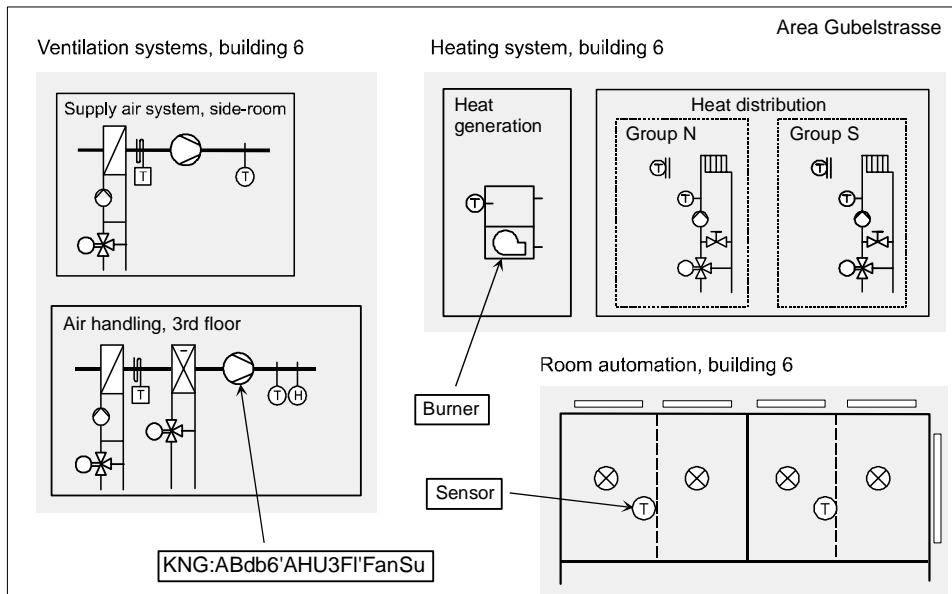
| Function | Description |
|---|---|
| Automatic control | Automatic switch-on/switch-off based on brightness, constant lighting control. In simply terms, this function achieves optimum lighting conditions automatically in occupied rooms, and switches off lighting when rooms are unoccupied. |
| Manual operation (room, central) | Manual operation allows users to themselves determine brightness via buttons. If manual operation overrides a lower-priority function, a scheduler program or local presence information will reactivate the function. |
| Presence-based influence (room) | Automatic switch-on when dark upon entering a room, and automatic switch-off when leaving a room. The presence-based function generally acts on the same priority as manual operation. |
| Scheduler program | Lighting can be switched on/off at specific times using a scheduler program. Furthermore, automatic control can be activated or deactivated via scheduler program. Another priority may need to be commanded depending on purpose. If, e.g., automatic control should be activated at noon, manual operation must be overridden by allowing the scheduler program to act on the priority for manual operation. If lighting is to be switched off at night without allowing for manual operation, a higher priority must be commanded. |
| Manual operation at high priority (room, central) | Manual operation at high priority allows for influencing lighting blinds and overriding low-priority functions. For example, this function allows for ensuring that neither motion detectors nor scheduler programs can switch on/off lighting at the wrong time during a lecture/presentation. |
| Maintenance, central | For maintenance or cleaning, lighting is switched on/off at high priority enabling staff to carry out all required work without risk of injury or being interrupted. |
| Protection, central | Lighting can be switched on in the event of a fire alarm to light escape routes or support emergency crew access. |

A very simple control contains just one or two functions, A complex plant may use many or all available functions. In addition, the response of individual functions may require parameterization depending on the requirements. The following figure shows an example of a plant including all functions.



6 Technical view

The technical view illustrates the technical building services equipment, such as HVAC systems and associated elements, in the building automation and control system.



The technical view helps organize measured and controlled physical variables from specific, technical installations in a building. The technical view is modeled with structure objects. The structure of the technical view represents the hierarchy of the technical installations. Objects representing variables, such as setpoints or operating modes supplement the view.

Plant types

The technical view contains all the conceptual objects in the system. The following plant types have been defined for descriptions and categories:

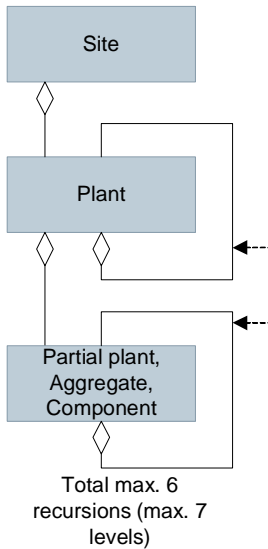
- *Primary plant:* All physical plants that are directly controlled from the automation level, e.g., heating systems, ventilation systems, etc.
- *Room automation:* Individual room controls.
- *Global objects:* Data objects which exist simultaneously in several automation stations at the automation level, e.g., an exception calendar for the time schedules of all plants. These objects are combined as a virtual plant in a global area and can be invoked as such (global data).

The technical view can be used for other disciplines integrated via PX Open. The technical view and the associated technical designations can be set up in the compounds library.

6.1 Standard plant structures

To display different plants in a uniform manner, a standard hierarchical plant structure has been created for each plant type.

Primary plants with Desigo PX Structure



Elements

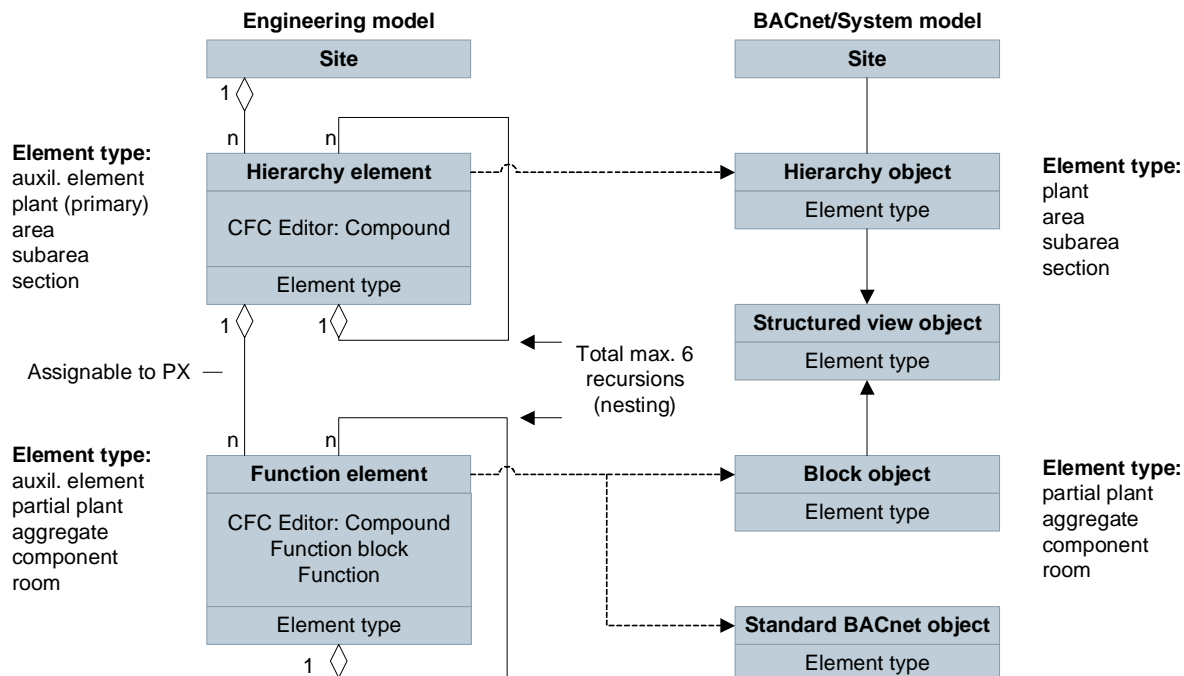
Site: A site is a self-contained area in terms of location, function and organization, usually a building or a group of buildings (facility). A site can comprise several plants. Example: Building 6

Plant: A plant consists of partial plants, aggregates and components. A plant can comprise several partial plants. Aggregates and components can be directly subordinate to a plant. Example: Ventilation system, heating system

Partial plant: A partial plant can comprise various aggregates. Components can be directly subordinate to a partial plant. Example: Central supply air treatment, air distribution, hot water supply (one or more boilers)

Aggregate: An aggregate can comprise various components. Example: Exhaust air fan

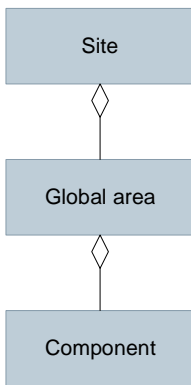
Component: A component can comprise several components, which can comprise several components themselves. Example: Pumps (motors), dampers, valves, sensors, detectors, limit switches, contactors, selector switches, remote/local switches



Element type: auxiliary element, plant, partial plant, aggregate, component, area, subarea, section, room

Global objects

Structure



Elements

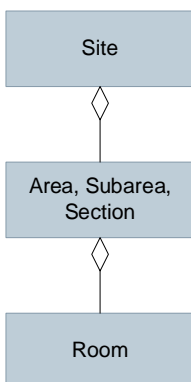
Site: A site is a self-contained area in terms of location, function and organization, usually a building or a group of buildings (facility). Example: Building 6

Global area: The global area contains all the global components of the site. There is one global area per site. Global objects are data objects which exist simultaneously in several automation stations at the automation level, e.g., an exception calendar for the time schedules of all plants. These objects are combined as a virtual system in a global area and can be invoked as such.

Component: A global area may contain several components, such as 3 calendars, 18 notification classes for alarm distribution. Each component is present on all automation stations of the site. For operation, however, each component is visible only once.

Room automation with Desigo RX

Structure



Elements

Site: A site is a self-contained area in terms of location, function and organization, usually a building or a group of buildings (facility). A site can comprise several plants. Example: Building 6

Area: An area is typically a building, and can comprise subareas, sections, components and subcomponents. Example: Building

Subarea: A subarea is typically the wing of a building and can comprise several sections. Rooms can be directly subordinate to a subarea. Example: Building wing, staircase

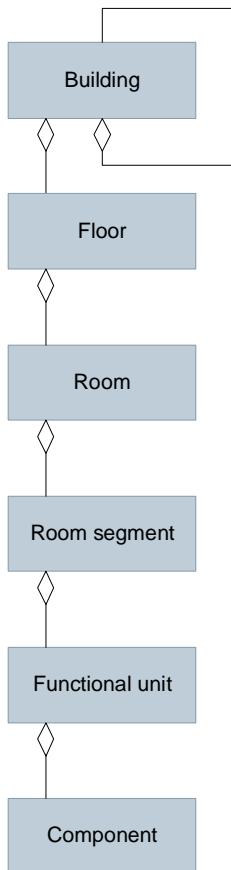
Section: A section is typically a floor in a building and can contain various rooms. Example: Floor

Software objects, which need to be displayed and operated even though they do not exist as physical elements in a real building, are also treated both as sections (e.g., via grouping criteria, such as east facade or emergency group 12) and as components (e.g., group object for distribution of centrally determined control variables to several rooms).

Room: A room is a section of a building that is delimited by walls, ceilings, floors, windows and doors. Example: Individual room, hall

Room automation with Desigo room automation

Structure



Elements

Building: A building is a locally, functionally and organizationally defined area. Example: Building 6

Floor: A floor in a building can contain various rooms. Example: Floor

Room: A room is a section of a building that is delimited by walls, ceilings, floors, windows and doors. Example: Individual room, hall

Room segment: A room segment is a subdivision of a room. A room can contain several room segments.

Functional unit: A functional unit is a logical component representing an encapsulated application unit which may be independently deployed to an arbitrary automation device. Example: Fan coil

Component: A functional unit can contain several components. Example: Awning

6.2 Technical text labels

The Technical Designation (TD) is a technical identifier that is used to identify the plant and associated elements.

The structure of the TD is based on the hierarchical structure of the plant and its associated elements, e.g.:

- For primary plants with Desigo PX:
Site / Plant / Partial plant / Aggregate / Component / Pin
- For room automation plants with Desigo RX:
Site / Area / Subarea / Section / Room
- For room automation plants with Desigo room automation:
Building / Floor / Room / Room Segment / Functional Unit / Component / Pin

The text is based on designations in abbreviated form that are customary within the industry, e.g.:

GUB:AGeb6'Ahu3St'FanSu = Gubelstreet facility / ventilation plans building 6 / Air handling third floor / supply air fan

Technical designations are linguistically neutral (mnemonic). They are based on mnemonic texts set up in the library, with additional project-specific details.

The TD is defined by Siemens. The User Designation (UD) can be defined by the customer.

Name&Description_Pair

Each element of the TD is called *ShortName*. A *ShortName* is a designation for an individual plant element within the automation station. A *ShortName* is always linked to a description. This pair is called the *Name&Description_Pair*.

TD rules

| Item | | Rule |
|-------------------|----------------------------|--|
| Address structure | | Comprises at least one hierarchical object and one function object Has a variable length (site + 1..8 hierarchy and function objects + pin name) Is independent of the automation station, that is, does not contain a designation for an automation station Must be unique for each site |
| Mnemonic | | Based on the English terms Must not be translated Plant elements on the same hierarchy level are distinguished through different part names, e.g., HG01/HG02. |
| Syntax | Site designation | Consists of upper and lower case letters, and numbers (must start with a letter): [a..z, A..Z, 0..9]. Is case insensitive, e.g., <i>Imax</i> and <i>IMAx</i> are treated the same. |
| | Other partial designations | Consist of upper and lower case letters ([A..Z] and [a..z]) and/or numbers 0 to 9. Is case insensitive, e.g., <i>Imax</i> and <i>IMAx</i> are treated the same. |
| | Max. number of characters | Site: 10 Object: 9 Pin: 8 TD: 80 |
| | Separators | Colon (:) after site name Apostrophes (') for all other separations Period (.) in front of pin name |

Function blocks and pins

A function block, that is, an object with pins, can represent an area, a partial plant, a sub-area, an aggregate, a section, or a component. Function blocks have attributes and function block pins have attributes.

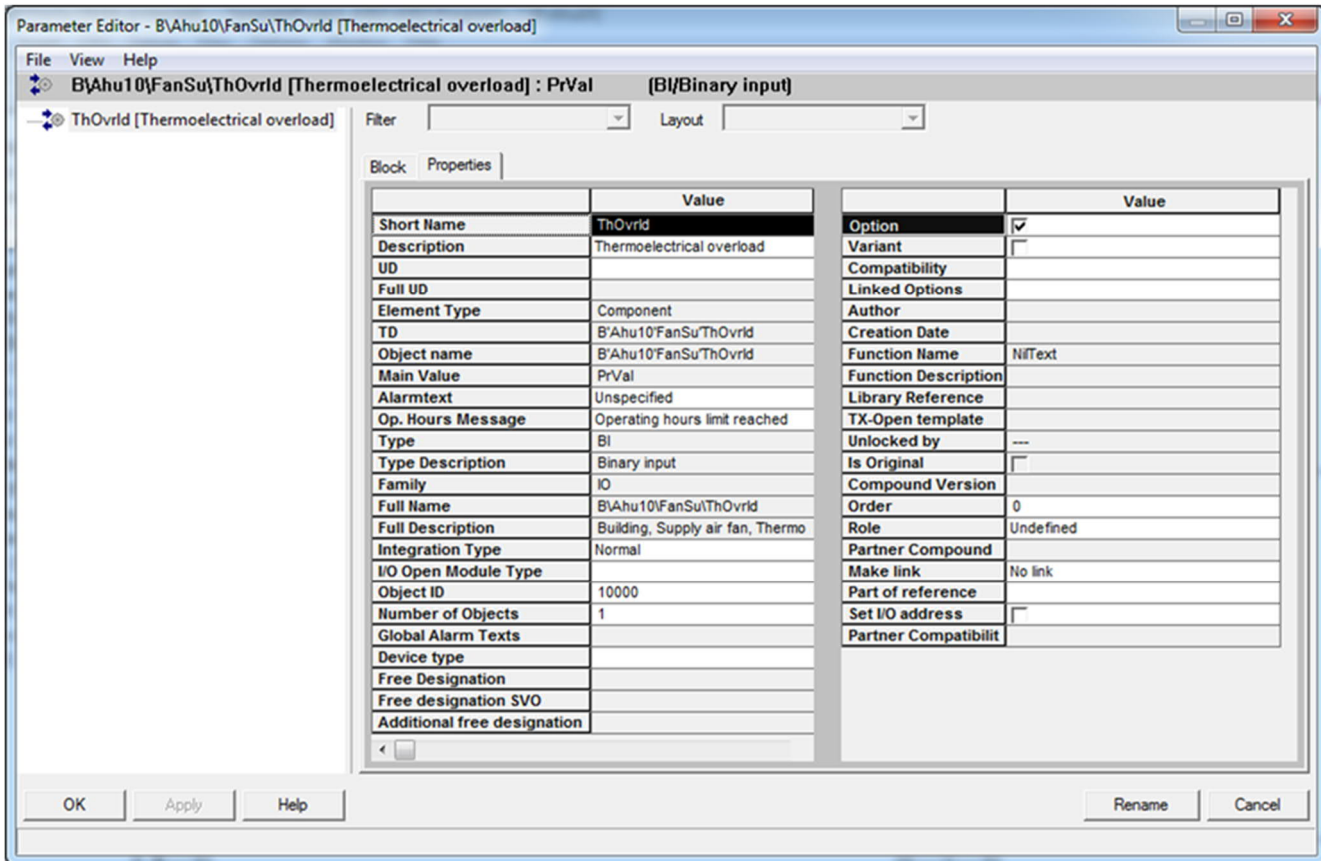
The following figure shows a function block and its pins as they appear in the program view:



Function block attributes

The main attributes of the function block are:

- *Name*: Name of the function block based on the key of the TD. Example: ThOvrld
- *Description*: Additional description. For generic operation it is shown as text in an operator unit. Example: Thermoelrical ovrl
- *Element type*: Block in plant-engineering terms. Example: Component
- *Main value*: Main value of the function block. It is set during engineering. Example: PrVal



Function block pin attributes

The main attributes of the pins are:

- *Name*: Pin name, based on the key of the TD. Example: PrVal
- *Description*: Description of the pin name. Example: Present value
- *Value*: Current value of PrVal. Example: Normal
- *Parameter Kind*: Application pin type. Example: Process input
- *Data Type*: Data type of the pin. Example: Multistate

For a complete list of attributes, see *CFC Online Help*.

Parameter Editor - B\Ahu10\FanSu\ThOvrd [Thermoelctrical overload]

File View Help

B\Ahu10\FanSu\ThOvrd [Thermoelctrical overload] : PrVal [BI/Binary input]

Filter: All pins Layout: All attributes

Block Properties

| Short Name | Description | Value | Parameter Kind | Data Type |
|------------|------------------------------|---------------------|----------------|------------|
| > AckSgn | Acknowledgement signature | | State | String |
| > AckTra | Acknowledged transitions | To offnormal, T | State | BitString |
| > AlmCl | Alarm class | High priority alarm | Alarm configu | Multistate |
| > AlmFunct | Alarm function | Basic | Alarm configu | Multistate |
| > ComgSta | Commissioning state | Not checked | Commissionin | Multistate |
| > Csc | Change of state count | 0 | Operation | Unsigned |
| > EnAlm | Alarm enable | Yes | General | Boolean |
| > EnCsc | Enable change of state count | No | Operation | Boolean |
| > EnEvt | Event enable | To offnorma | Adaptation | BitString |
| > EnOph | Enable operating hours count | No | Adaptation | Boolean |
| > EvtSta | Event state | Normal | State | Multistate |
| > IOAddr | Input/Output address | P=0.(D20) | Commissionin | NameRe |
| > MntnDate | Maintenance date | ---:---:--- | Operation | Date |
| > NotifCl | Notification class | | General | Unsigned |
| > NotifSel | Notification funct.selector | Predefined | General | Boolean |
| > NotifTyp | Notification type | | Alarm configu | Multistate |
| > OoServ | Out of service | Off | Operation | Boolean |
| > Oph | Operating hours | 0000000:00:00 | Operation | Duration |
| > OphLm | Op.hours limit value | 0000000:00:00 | Operation | Duration |
| > OphTot | Total operating hours | 0000000:00:00 | Operation | Duration |
| > Pol | Polarity | Reverse | Advanced co | Boolean |
| > RefVal | Reference value | Alarm | Alarm configu | Boolean |
| > RibDly | Reliability delay | 00:00:00 | VO configurat | Duration |
| > SbstVal | Substitution value | Normal | VO configurat | Boolean |
| > SupEcpt | Suppress event exception | No | Alarm configu | Boolean |
| > TiAck | Time of acknowledgement | --- | State | DateTime |
| > TiMonDvn | Monitoring time deviation | 00:00:00 | Alarm configu | Duration |
| > TiMonOff | Monitoring time switch-off | 00:00:00 | Alarm configu | Duration |
| > TiMonOn | Monitoring time switch-on | 00:00:00 | Alarm configu | Duration |
| > TPin | Poling time | 00:00:00 | Commissionin | Duration |
| > TiStmCsc | Time of state count reset | --- | Operation | DateTime |
| > TiStmEvt | Time stamp of event | --- | State | Array |
| > TiStmOph | Time stamp for op hour reset | --- | Operation | DateTime |

| Short | Description | Value | Parameter Kind | Data Type |
|------------|------------------------------|-----------------|----------------|------------|
| < Dstb | Disturbance | No | State | Boolean |
| < Rib | Reliability | No fault detect | State | Multistate |
| < StaFig | State flag | --- | State | BitString |
| < PrVal | Present value | Normal | Process input | Boolean |
| < SupActv | Event suppression active | No | Process input | Boolean |
| < EvtMsg | Event message texts | --- | Operation | Array |
| < PrOphLm | Pres.op.hours limit | 0000000:00:00 | Operation | Duration |
| < PrOph | Pres.operating hours | 0000000:00:00 | Operation | Duration |
| < MntnInd | Maintenance indication | No | Operation | Boolean |
| < PrCsc | Present change of state coun | 0 | Operation | Unsigned |
| < CosTi | Change of state time | --- | Operation | DateTime |
| < PrOphTot | Pres.total op.hours | 0000000:00:00 | Operation | Duration |

OK Apply Help Rename Cancel

7 Global objects and functions

Every automation station contains all the data necessary for stand-alone operation, including, e.g., date and time, calendar function blocks and Notification Class function blocks. The system functions of individual automation stations do not depend on a central server.

The System View and the Program View are based on the automation station, that is, each object (block, BACnet data object) belongs to a specific automation station. These objects are called local objects. This form of representation is adequate for most elements of a physical plant, e.g., for the supply air temperature or the set point of a ventilation system.

However, certain data objects need to be visible in identical form in some or all the automation stations of a site. These objects are called global objects. Global objects let you centrally change parameters, which are then distributed to all automation stations.

Local objects

Local objects are individual and unique objects which exist only once on a particular automation station in the system. Most application-related objects are local objects. When local objects are required, such as the outside temperature in several automation stations, access to this data must be configured or programmed explicitly with function blocks (such as analog, binary and multistate inputs, or grouping in the room management system) and referencing.

Global objects

Global objects are data objects which exist simultaneously on each automation station at the automation level. Global objects are always global within a given site. Global objects are engineered in Xworks Plus (XWP).

Global objects are compiled in a global chart. There is exactly one global chart per site. You can modify global charts, save them in the tool's library folder, and copy them to other projects.

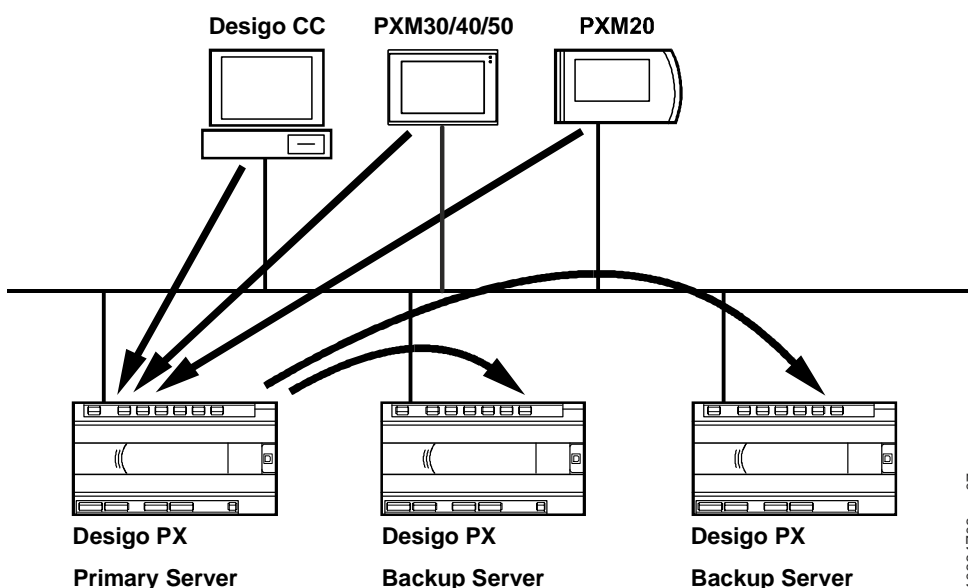
7.1 Ensuring data consistency

Primary copy

The primary copy procedure ensures that the global objects are consistent at all times. This means that all copies of a particular global object contain the same value and any modification of a value is transmitted to all copies.

Primary and backup server

Only one automation station per site acts as the primary server for all global objects of this site. All other automation stations of this site are backup servers. A client may only modify the values of the global objects on the primary server. The primary server then updates the copies of the modified global objects on all backup servers. A backup server accepts the modifications to global objects only from its primary server but not from a client.



Xworks Plus (XWP) and all BACnet clients can only modify the data of global objects in the primary server.

7.2 Roles in the system

The role of the primary server

| Server/Function | Function and description |
|----------------------------|--|
| Primary server (Desigo PX) | One automation station of a site acts as the primary server. Make sure that only one primary server exists at any one time on a site. |
| Life check | The primary server monitors the backup servers and the third-party BACnet devices of a site. The primary server can monitor the Desigo room automation server. |
| Time synchronization | The primary server synchronizes the time of the backup servers and the third-party BACnet devices of the site. The primary server can synchronize the Desigo room automation server time. |
| Start-up | No coordinated start-up. |
| Replication | The primary server replicates the global objects and properties (device object) to the backup servers of a site. A backup server accepts changes of global objects only from the primary server. |

The role of the backup server

| Server/Function | Function and description |
|---------------------------|--|
| Backup server (Desigo PX) | The other automation stations of a site must be backup servers. |
| Life check | The backup servers monitor the primary server of the site. Backup servers can monitor Desigo room automation servers. |
| Time synchronization | The backup server can synchronize the time of the Desigo room automation server and of the third-party BACnet devices. |
| Start-up | No coordinated start-up. |

The role of the Desigo room automation server and third-party BACnet device

| Server/Function | Function and description |
|---|--|
| Desigo room automation server / Third-party BACnet device | The Desigo room automation server acts like a standard BACnet device. |
| Life check | The Desigo room automation server / third-party BACnet device is monitored by the primary server or the backup server. |
| Start-up | No coordinated start-up. |
| Replication | No global objects to be replicated. |

The role of the clients

| Server/Function | Function and description |
|-----------------|---|
| Clients | A client can read global objects from the primary server or any backup server. A client may only modify a global object (e.g., with WriteProperty) on the primary server. Each client must therefore recognize the primary server of a site. A client can query the identification of the primary server of a site. Replicated objects from backup servers which are not online or which are connected to the BACnet network at a later stage, are updated by the primary server as soon as the primary server becomes aware of them. This occurs after a restart of the automation station, after connecting it to the network or on expiry of the synchronization request period [SynReq]. |

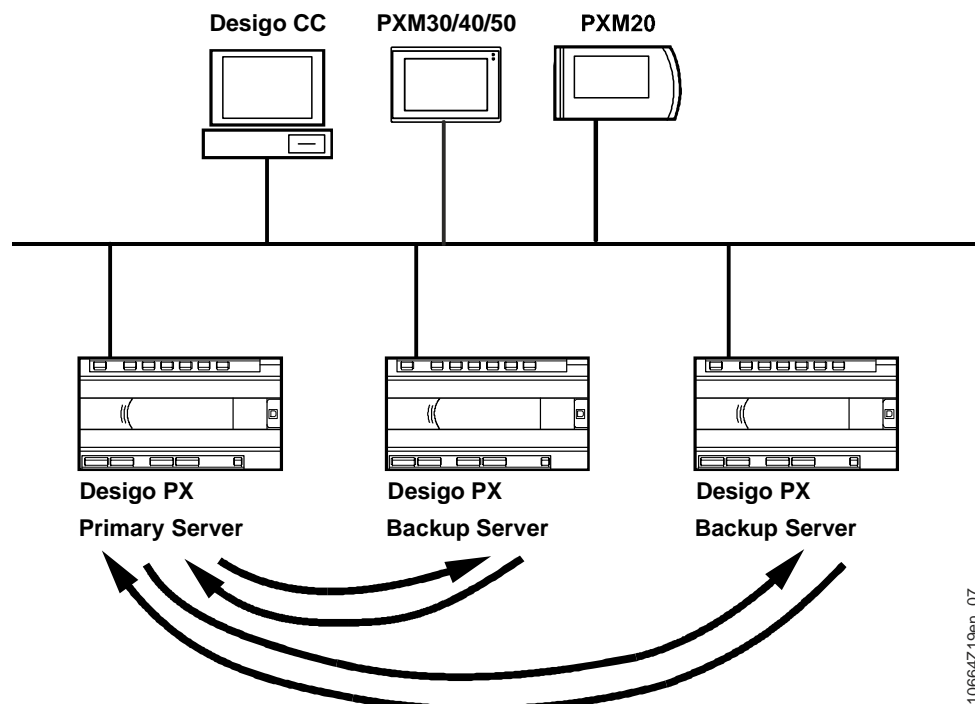
The role of the alternative primary server

| Server/Function | Function and description |
|----------------------------|---|
| Alternative primary server | If the primary server fails, no global objects can be modified. You can configure any backup server of the site to act as the primary server using a client or Xworks Plus (XWP). |

7.3 Life check

The life check checks if all devices of the site (primary server, backup server, Desigo room automation server or third-party BACnet device) work correctly, that is, if they are operating and if they are running their application.

- The primary server monitors if the backup servers / Desigo room automation servers are active.
- The backup servers monitor if the primary server is active.
- The primary server checks that only one primary server exists at any one time on a given site.



Add and delete devices

For life check and replication the primary server has a list [BckUpSrv] of all known devices of a site. The primary server automatically adds newly commissioned devices on the site to this list. Devices which are removed from the site must be deleted manually in Xworks Plus (XWP) from the list in the primary server.

Check if all devices are online

The primary server performs a life check at regular intervals, to check that all devices of its site are online. The interval between life checks is defined by the synchronization request period [SynReqp]. During this period, the backup servers are checked one after the other in a cyclical process. The interval between two life checks can be calculated as follows: $t \approx \text{SynReqp} / \text{Number of backup servers}$.

A short synchronization request period and a large number of backup servers may involve a substantial communications load. Take this into account when setting the synchronization request period in Xworks Plus (XWP).

If one or more devices are not online, the primary server generates an alarm signal. The alarm is reset as soon as all devices are online again and have been detected by the primary server. This ensures that problems, such as the failure of a device, the termination of the HVAC-application processing of a device, or faulty configuration (e.g., two primary servers in one site) are detected.

Monitor if the life check checks the backup server

Each backup server monitors its own periodic life check by the primary server. If a life check fails, or if no primary server has ever carried out any life checks on this backup server, the backup server generates an alarm. The backup server resets the alarm as soon as the primary server carries out a life check.

7.4 Time synchronization

Each automation station is assigned to a site.

The primary server is the time master. It represents the system time within a given site. The primary server synchronizes the time in the other automation stations at regular intervals.

If the primary server receives a time synchronization request which triggers a time change, the primary server synchronizes the time in the other automation stations.

The primary server transmits the time in UTC format (Coordinated Universal Time) to the other automation stations (backup servers) and in UTC format or local time format to the third-party BACnet devices.

The backup server then triggers time synchronization of its recipients configured in Xworks Plus (Desigo room automation server, third-party server, third-party BACnet devices). This can be in either UTC or local time format.

Periodic synchronization

The time synchronization interval is defined in the property TimeSynchronizationInterval [TiSynIvl] (default value: 150 minutes). The property can be configured in Xworks Plus (XWP) and adapted to the specific situation via a switchable [AlgnIvl] offset [IvlOfs]. How these three properties function is defined in the BACnet standard and implemented accordingly.

Add and delete devices

For time synchronization, the primary server has a list [TiSynRcp] containing the recipients configured in Xworks Plus (XWP) and all known backup servers of its site.

The primary server automatically adds newly commissioned backup servers on the site to the list [TiSynRcp].

Backup servers which are removed from the site must be removed manually from the primary server list in Xworks Plus (XWP).

Link the system time of a site with operator units

The network-capable operator units do not belong to a site. The primary server does not update the time in the operator units. The client can read and update the time, if required.

Daylight saving time changeover

The daylight saving time changeover does NOT take place in the primary server. Each automation station makes this switch independently.

The date of the daylight saving changeover is set as a parameter in the primary server. The primary server replicates the date on the backup server. The official (Central European) changeover date is set as default.

The local time in an automation station is a calculated variable. The calculation is based on the internal time in UTC format, the property UTC offset [UtcOfs] and the date of the summer and winter time changeover.

See *Desigo CC User Guide* (A6V10415471).

7.5 Examples of global objects

BACnet device object

Certain properties of the BACnet device object are defined as global, because from the perspective of the system, they are required to have the same value throughout the site. These properties are set in Xworks Plus (XWP). Examples:

Global properties

- Date and time for the summer and winter time changeover.
 - Daylight savings time start date [DsavSdt] (Default: last Sunday in March)
 - Daylight savings time start time [DsavSti] (Default: 02:00AM)
 - Daylight savings time end date [DsavEdt] (Default: Last Sunday in October)
 - Daylight savings time end time [DsavEti] (Default: 03:00AM)
- UTC_Offset [UtcOfs]
 - Difference between UTC and local Winter time in minutes. Default value: – 60 mins (Central Europe). In Summer, the effective difference [UtcOfs] is –60 mins (Central Europe: –120min).
- Synch.Request Period [SynReqp]
 - Period between life checks by primary server The load on the communications system generated by the life check can be controlled with this parameter by adapting it to the site size. Default value: 1800 s.

- Name resolution interval [NamRI]
Periodic repetition for the resolution of references across devices. Default value: 900 s.
- COV resubscription interval [CovRI]
Time within which the automation station resubscribes to a subscribed value. Default value: 1800 s.

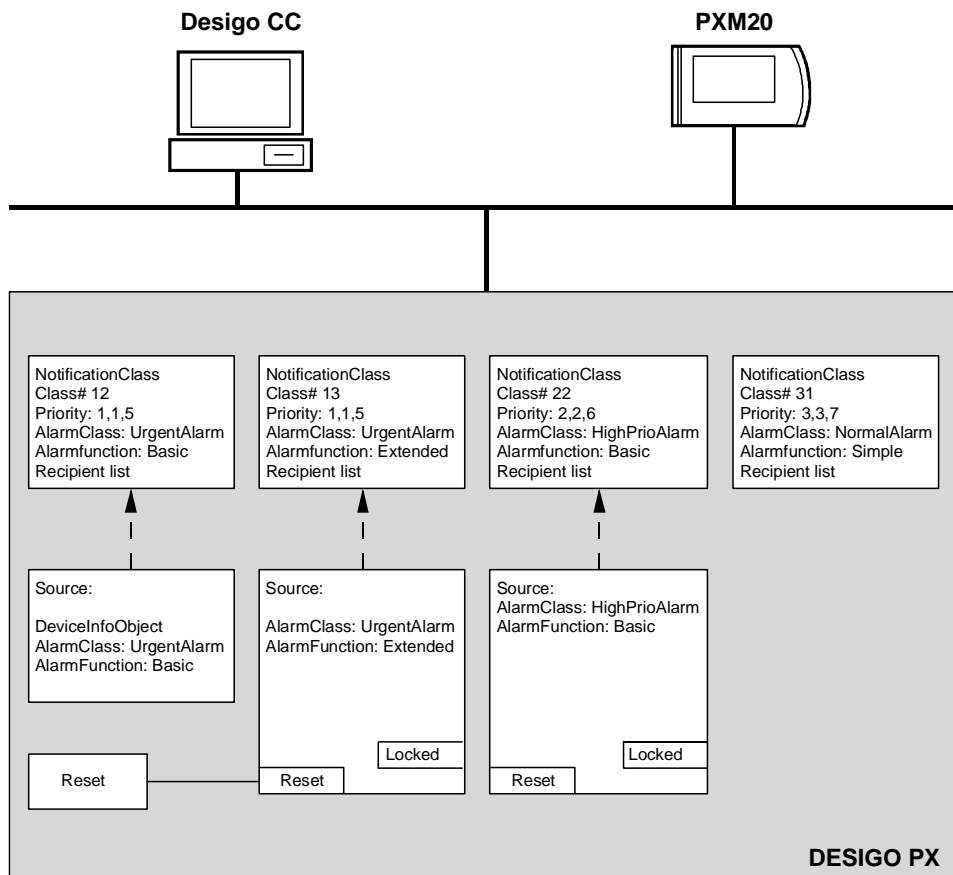
Local properties

Local properties which refer to the functionality of the life check / replication:

- Server type [SvrTyp]
Defines if the device acts as a primary server or a backup server. Default: backup.
- Primary device [PrimDev]
Device object ID of the primary server of the site or an invalid value if the primary server is not known (read-only, set automatically by the primary server).
- Last engineering time, global object [GOEngTi]
Time stamp of the last structure modification of the global objects by Xworks Plus (XWP).
- Last online modification of global objects [GOChgTi]
Time stamp of the last online modification of a global object in Xworks Plus (XWP) (modified by the primary server, read-only).

Notification class object

The Notification Class object is a standard BACnet object and defines the system response of alarms and system events.



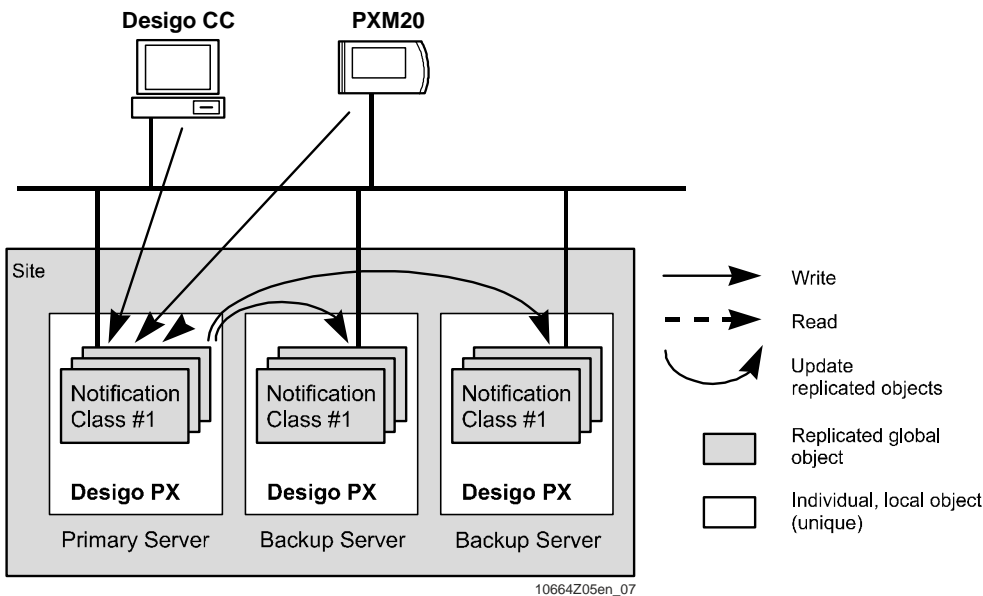
There are local and global Notification Class Objects.

Global Notification Class Object: A logical object at site level that exists in identical form (as a replicated object) in every automation station on a site.

Local Notification Class Object: Individual object (unique object) that exists only on a particular automation station.

Reading of objects by a client: A client may read the global notification class objects from any automation station.

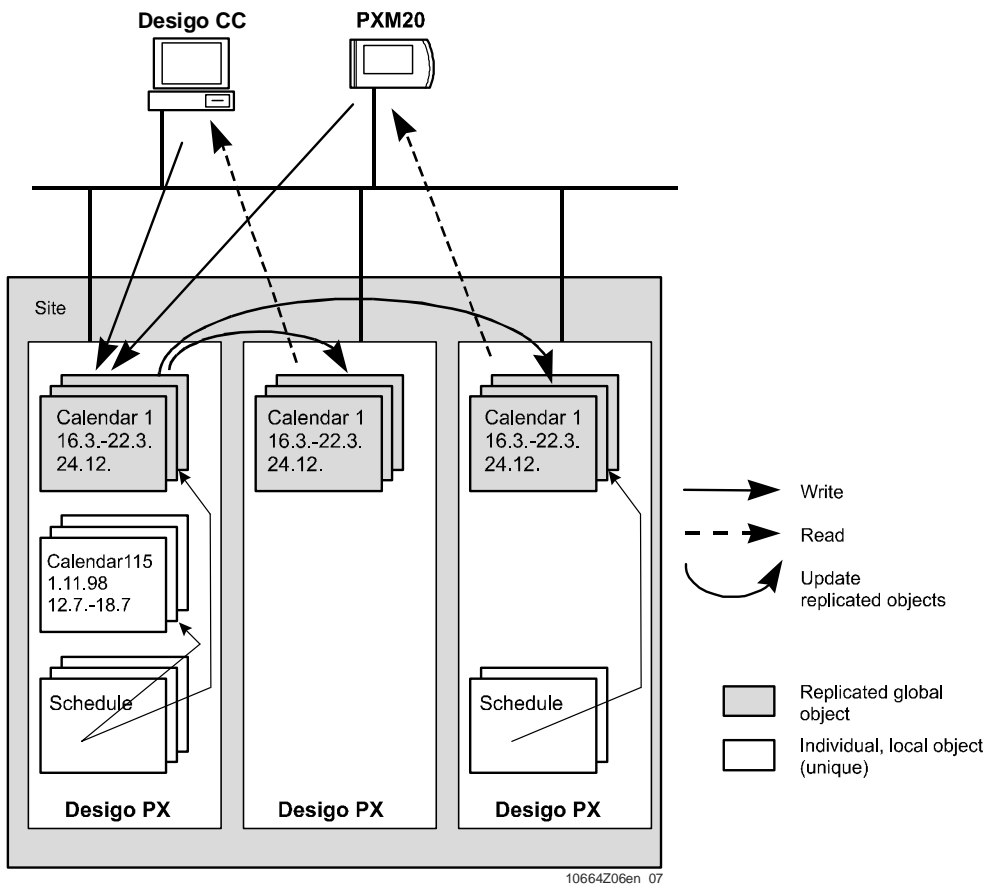
Reasons for replication: Keeping the setting parameters consistent for all automation stations of a site when modifications occur (adding or deleting configured recipients from recipient list, changing priorities).



The number of global Notification Class objects is limited to 18 (six alarm classes each with three possible alarm functions).

Calendar object

There are global and local calendar objects.



Global calendar object: A logical object at site level. It exists in identical form (as a replicated object) on each automation station of a site.

Local calendar object: Individual (unique) object that exists only on a particular automation station.

Local processing: Schedule objects in an automation station may reference the replicated calendar objects in the device. A client may read the global calendar objects from any automation station.

Reasons for replication: Global exceptions (bank holidays, general holidays, etc.) can be modified centrally in one location for the entire site. Ensures continuity of operation if the master fails.

User profile object

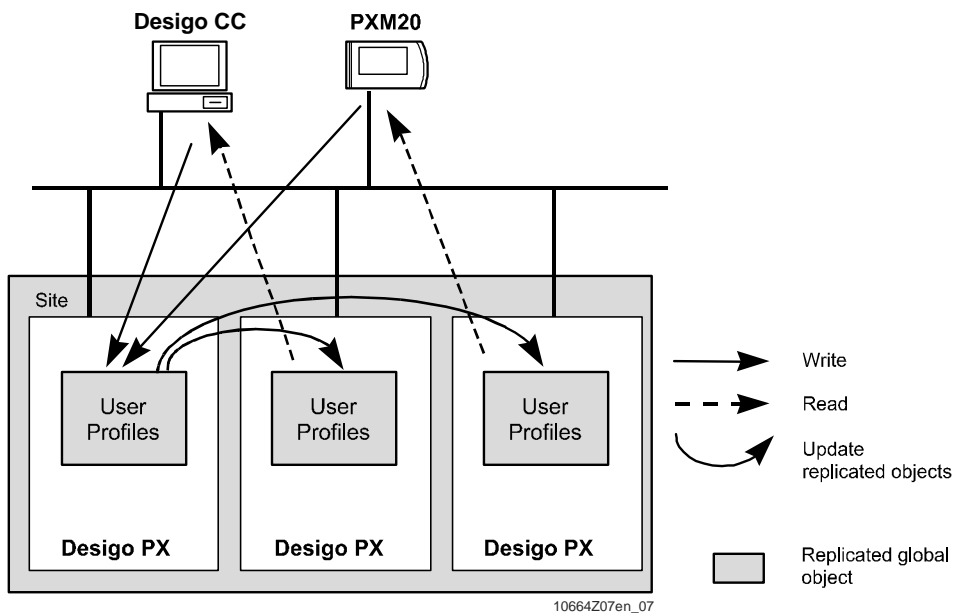
Global user profile object: A logical object at the site level. It exists in identical form (as a replicated object) on each automation station of a site. There must be at least one user profile object.

There are no local user profile objects.

Local processing: Access control is based on the replicated user profile objects in the automation stations (BACnet devices): No dependency on a server.

Reading of objects by a client: A client may read the global user profile objects from any automation station.

Reasons for replication: Replication is designed to maintain consistency of the access rights throughout the site, and to ensure continuity of operation in the event of the failure of the master.



8 Events and COV reporting

Events

System events are messages which inform a client (e.g., Desigo CC) of specific events in an automation station, such as:

- Change in operating state of the automation station (STOP, RUN)
- Overflow of the operating hours counter built into certain I/O objects

Conceptually, system events are similar to alarms, however, they differ from alarms in some ways:

- System events have no memory, that is, they do not have a finite state machine.
- System events do not affect the status of a plant, that is, they can occur in any alarm state without influencing it.
- System events are displayed, but do not need to be acknowledged or reset.

System events are forwarded to clients using the same mechanism as alarms.

COV reporting

If a value of a specific process variable changes, the change is transmitted to other system components by means of Change Of Value (COV) reporting. Polling is used only in exceptional cases. COV reporting can be used to transfer value changes to several automation stations. A COV notification is issued only when the value of the process variable changes in comparison with the preset or default incremental value. There is no need to poll the process variables at regular intervals.

There are two roles:

- COV-Server: The automation station which reads the process variable and whose change of value is to be reported.
- COV-Client: The recipient of the COV notifications. This may be another automation station or a BACnet client.

8.1 Sources and causes of system events

The source of a system event is a function block (as with alarms). System events can originate from the same block types as alarms:

- Analog Input/Output/Value
- Binary Input/Output/Value
- Multistate Input/Output/Value
- BACnet Device Info Object (not in Desigo room automation)

Every block type capable of generating a system event has a clearly defined set of system event triggers.

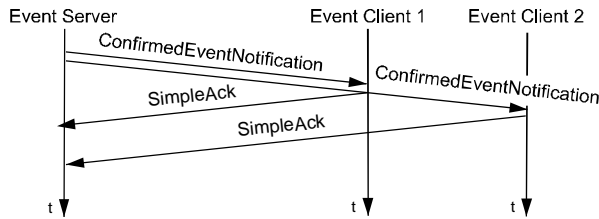
| Event-generating block types | Description |
|------------------------------|--|
| Operating hours counter | The input, output and value objects of the Binary and Multistate types have an inbuilt operating hours counter. A system event is generated when the operating hours limit is exceeded or when the maintenance interval has expired. |
| BACnet Device Info Object | The BACnet Device Info Object detects the causes of system events which apply to the automation station as a whole. The following causes are detected: <ul style="list-style-type: none"> - Change of operating state (start and stop the program) - Restart after a power-up - Primary server has found a new backup server on the network - Backup server has found the primary server |

8.2 Routing system events

System events are forwarded to clients using the same mechanism as alarms. They are forwarded to all temporary and configured alarm recipients in accordance with the settings in the associated Notification Class objects.

Comparison with the alarm strategy

System events cannot be acknowledged or reset. A Confirmed Event Notification message is sent to all alarm recipients. The Notify_Type data field in the message defines that the event is a system event and not an alarm. Each alarm recipient that receives the Confirmed Event Notification is required to respond with a SimpleAck. If the SimpleAck is not received, the same mechanism comes into operation as for alarms.



Event texts

Each system event has a message text assigned to it. For the system events related to the operating hours counter, a user-specific text can be set up in Xworks Plus (XWP). Predefined system texts are available for the other system events.

8.3 Sources and causes of COVs

Process variables which can be mapped to standard BACnet objects are COV-capable.

I/O function block

Function blocks for Analog, Binary and Multistate Inputs, Outputs and Values are mapped directly to the associated BACnet object types. They are COV-capable and can establish COV connections with all COV clients.

Interface variables

Interface variables of compounds and function blocks whose data type is Analog, Binary, Multistate, Integer, Duration and DateTime are COV-capable and can be mapped to simplified BACnet value objects for operation and monitoring.

A COV is initiated when the value of the process variable [PrVal] of the BACnet object which represents it changes. A COV is also initiated when a status flag [StaFlg] (InAlarm, Fault, Overridden or Out of service) changes, e.g., when a sensor open circuit (fault) occurs or when an I/O value is overwritten manually.

COV increment

For analog objects, a COV is not initiated for every minor change of [PrVal], but only when the value changes by an amount greater than a predefined increment. This increment is saved in the COV increment [COV] of the analog object, and can be defined in Xworks Plus (XWP) during engineering.

8.4 COV reporting

Subscription

Each COV client must subscribe to every process variable from which it requires COV notifications. Each COV-capable object transmits COV notifications only to those COV clients which have subscribed to COV notifications. The subscription process is carried out using the BACnet service SubscribeCOV, transmitted by the COV client to the COV server. This message contains all the information that the COV server needs to send the COV notifications to the correct destination. It also includes a time period which determines the validity period of the subscription. The time period may be infinite.

For system limits, see chapter *System Configuration*.

COV notifications

The COV server reports every COV individually to each COV client which has subscribed to it. The BACnet service ConfirmedCOVNotification is used for this purpose. It contains the values of [PrVal] and [StaFlg]. The service is a Confirmed Service, which means that the COV client must acknowledge the notification (SimpleAck). This ensures that when a COV client ceases to be available, this will be recognized by the COV server. If no SimpleAck message is received, the transmitting device tries to send the information again (three times).

For system limits, see chapter *System Configuration*.

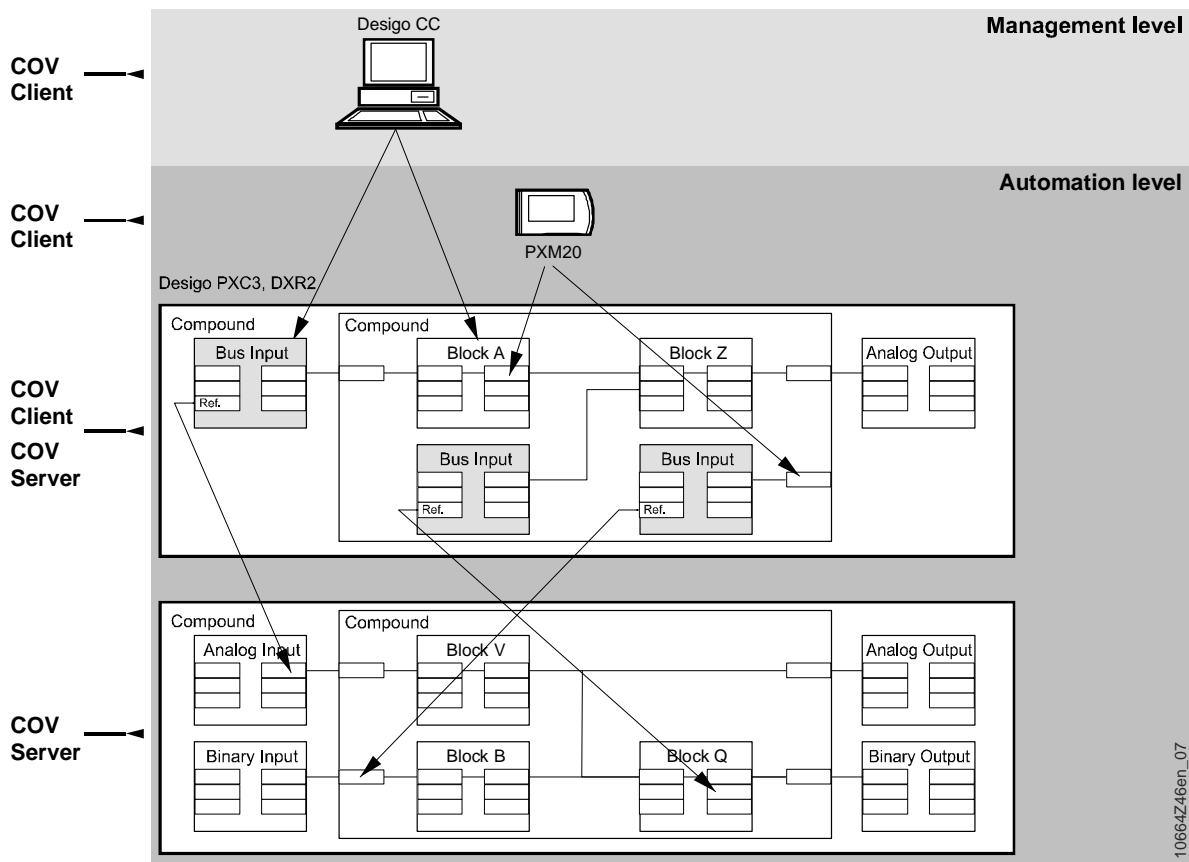
Connection terminated

If a COV client cannot be contacted, the COV server ceases to send COV notifications to that client. The transmission of COV notifications to a COV client is resumed when the COV client re-subscribes.

Checking the connection

To ensure that the COV service is maintained over a long period, a maximum time period without COV reporting can be set in the BACnet Device Info Object via the BACnet property COV Resubscription Interval [CovRI]. The client must subscribe with SubscribeCOV again before [CovRI] expires.

COV clients and COV servers

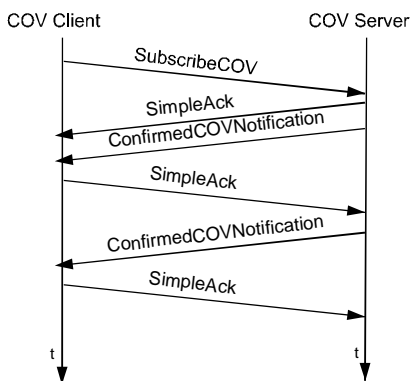


The local PXM10 operator unit is not a BACnet client and cannot, therefore, be used as a COV client. See *PXM10 operator unit: User's guide* (CM110397).

COV reporting between COV client and COV server

COV mechanism

BACnet clients use the COV mechanism for continuous monitoring of process variables without putting an excessive load on the bus through continuous polling. They subscribe to the objects that they are monitoring. These COV connections must be maintained as long as the object is being monitored.



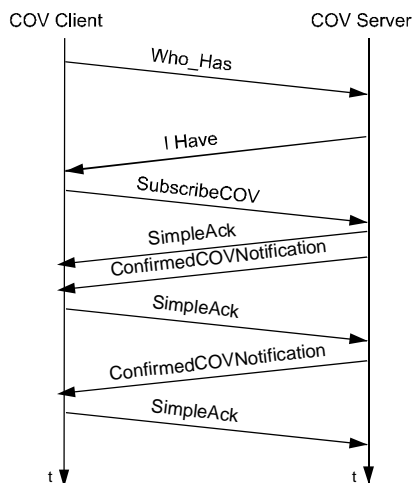
The BACnet client subscribes to the COV server as a COV client using the BACnet service SubscribeCOV. The server sends a SimpleAck acknowledgement. Immediately after the acknowledgement, the COV server transmits an initial ConfirmedCOVNotification. The COV client acknowledges receipt of the value with a SimpleAck acknowledgement. The COV connection between the COV server and COV client is now established, and ConfirmedCOVNotifications are sent whenever a trigger for the subscribed COV occurs.

The BACnet service `SubscribeCOV` includes a time limit for the COV connection. However, the COV client re-registers with the COV server before this limit expires, thus ensuring that the connection is maintained. A COV connection ends when the subscription period expires and is not renewed, or when the COV client can no longer be contacted, causing the COV server to stop sending notifications.

In addition to the `SubscribeCOV` service, a `SubscribeCOV Property` service is implemented, e.g., for the operation of plant graphics in Desigo CC. This enables the system to respond with appropriate speed to changes in the high or low limit.

COV reporting between automation stations

COV connections between automation stations are used to implement pre-engineered references, that is, for the exchange of process values between individual plant parts on different automation stations. In this case the receiver is an input function block of the relevant data type (Analog, Binary, Multistate). The input function block contains the technical designation of the required COV source in its input/output address parameter [IOAddr]. These COV connections must be permanently live. The COV mechanism enables a dropped COV connection to be re-established.



When an automation station connects, the BACnet service `WhoHas` searches the entire network for the object referred to in the COV client. The automation station concerned responds to the COV client with the BACnet service `IHave`. If the COV client cannot find the COV server, it repeats the `WhoHas` request after the time period defined in the BACnet Device Info Object Property Name resolution interval [NamRI] until the COV server is found.

The COV client registers for a limited period as a COV client with the COV server using the BACnet service `SubscribeCOV`. The server sends a `SimpleAck` acknowledgement. The value is then sent to the COV client for the first time by the COV server, using the BACnet service `ConfirmedCOVNotification`. The COV client acknowledges receipt of the value with a `SimpleAck` acknowledgement. The COV connection between the COV server and the COV client is established from this point on. According to the global property COV renewal interval `CovRI` of the BACnet Device Info Object, the COVsubscription is renewed. The lifetime used for `SubscribeCOV` is twice the COV renewal interval `CovRI`. The COV connection ends when the subscription period expires and is not renewed, or when the COV client can no longer be contacted, causing the COV server to stop sending notifications.

9 Alarm management

Alarms indicate faults in the HVAC plant and building automation and control system, and let you initiate corrective action, where appropriate. The management of alarms (generation, signaling, acknowledgement) is in compliance with the BACnet standard.

There are two alarm types:

- OFFNORMAL
- FAULT

OFFNORMAL

OFFNORMAL alarms (process alarms) occur when a process variable assumes an inadmissible value. What is inadmissible is determined during engineering. The relevant parameters are stored in all alarm-generating objects. An OFFNORMAL alarm always indicates a fault in a plant, while the automation system itself works properly.

Examples of OFFNORMAL alarms:

- Temperature in HTHW circuit is too high or too low
- Alarm generated by fire detection system
- A damper-motor feedback signal has not been received
- A time schedule cannot execute a command

FAULT

FAULT alarms are faults in the automation system itself (internal alarms). You cannot define the cause of a FAULT alarm during engineering. Nor is it possible for the user to suppress or otherwise influence the monitoring of FAULT alarms. FAULT alarms are intrinsically linked to the system. A FAULT alarm always takes precedence over an OFFNORMAL alarm from the same alarm source, because in the case of a FAULT alarm, there is some uncertainty about the reliability of the alarm source.

Examples of FAULT alarms:

- Faulty sensor (open circuit, short circuit, etc.)
- Buffer for storage of non-volatile data full
- Access to an I/O module failed
- Bus open circuit (RX integration)

Alarm detection procedure

Every alarm (OFFNORMAL or FAULT) can be uniquely allocated to a source. The alarm monitoring system is based on the principle of Intrinsic Reporting or Algorithmic Reporting as defined in the BACnet standard.

Intrinsic reporting

Intrinsic reporting means that alarm monitoring (target-actual comparison) takes place within the alarm-generating object itself (the alarm source). For this purpose, the function block contains the entire alarm state machine. Alarm detection does not require any function blocks with external functions. The alarm behavior of the object is defined by setting variables in the alarm-generating object (function block).

Algorithmic reporting

Algorithmic Reporting means that alarm suppression (target-actual comparison) occurs outside the alarm source. The alarm state machine is not located in the function block of the alarm source. For alarm detection, function blocks with external functions are required. The object's alarm response is not parameterized using variables of the monitored object (function block).

9.1 Alarm sources

The following function blocks can be alarm sources:

- Analog Input / Analog Output / Analog Value
- Binary Input / Binary Output / Binary Value / Pulse Converter
- Multistate Input / Multistate Output / Multistate Value
- Event Enrollment
- Command Control object²
- Power Control object²

- Schedulers (Analog / Binary / Multistate Scheduler object)²
- AlarmCollection object
- Discipline I/O^{1, 2}
- Trend Log / Trend Log Multiple
- Group^{1, 2}
- Device Info object, which models the properties of an automation station as a complete entity
- Loop object

Key

¹ Discipline I/Os, Groups, Time Scheduler and Trend Log Multiple support only system alarms, that is, only alarms of the FAULT type. Both function blocks can transmit more than one system alarm. The parameters [Rib] and [MsgTxt] provide detailed information about the cause of the most recent alarm message. The messages are transmitted in the order in which they occur, irrespective of the importance of the alarm.

² These function blocks only exist in Desigo PX.

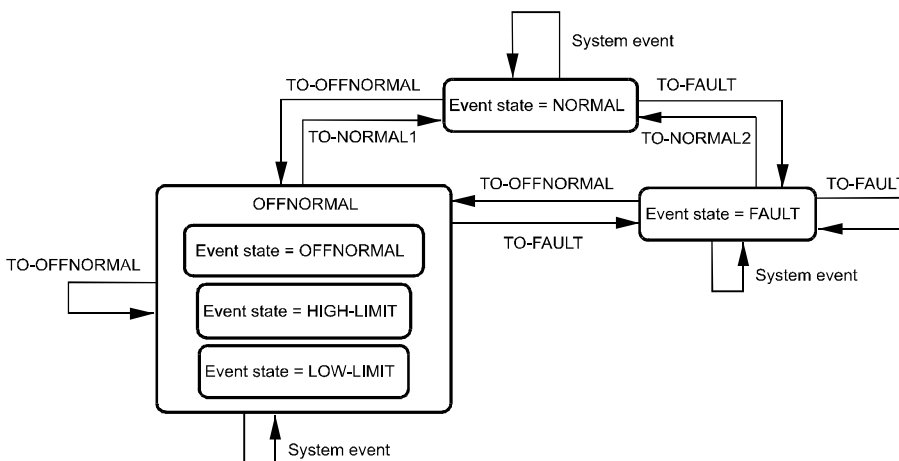
Only these alarm sources incorporate Intrinsic Reporting, and can thus generate their own alarms. If any other value of a function block needs to be monitored for an alarm (e.g., the control signal for a controller block), an Event Enrollment object must be added.

Alarm-generating function blocks include a range of interface variables which can be set as parameters to determine the alarm response (Input Property) or to supply the relevant alarm state information (Output Property). These interface variables are described further below. Some of the interface variables are common to all alarm-generating block types, while others are specific to certain types of alarm-generating blocks.

Alarm state machine in an alarm-generating function block

Alarm state machine

The response in the event of an alarm is modeled by an alarm state machine. Each alarm-generating block incorporates an alarm state machine of this type. The alarm-related interface variables can therefore be used to define the response of this state machine, to simulate state transitions, or to represent the current status of the state machine itself.



Alarm state event states

The alarm state machine can assume one of three basic states (event states [EvtSta]):

- NORMAL: There is no alarm condition present
- OFFNORMAL: Alarm caused by an OFFNORMAL condition
- FAULT: Alarm caused by a FAULT condition

With analog blocks, the OFFNORMAL state is explicitly subdivided into the sub-states HIGH LIMIT and LOW LIMIT, which are described in detail further below.

The current state of the alarm state machine in an alarm-generating block is displayed externally in the form of the output variable [EvtSta] (event state) of the block concerned.

State transitions between alarm states

| Transition | Trigger | Action / Event state |
|--------------|---|----------------------|
| TO_OFFNORMAL | A new OFFNORMAL alarm condition has been detected. | OFFNORMAL |
| TO_NORMAL1 | The current OFFNORMAL alarm condition has disappeared, and there is no other alarm condition present. | NORMAL |
| TO_FAULT | A new FAULT alarm condition has been detected. | FAULT |
| TO_NORMAL2 | The current FAULT alarm condition has disappeared, and there is currently no other alarm condition. | NORMAL |

System events may also occur within each alarm state. These message functions do not affect the alarm state. Because FAULT alarms take priority over OFFNORMAL alarms, the state transition from FAULT to OFFNORMAL only occurs under very special circumstances.

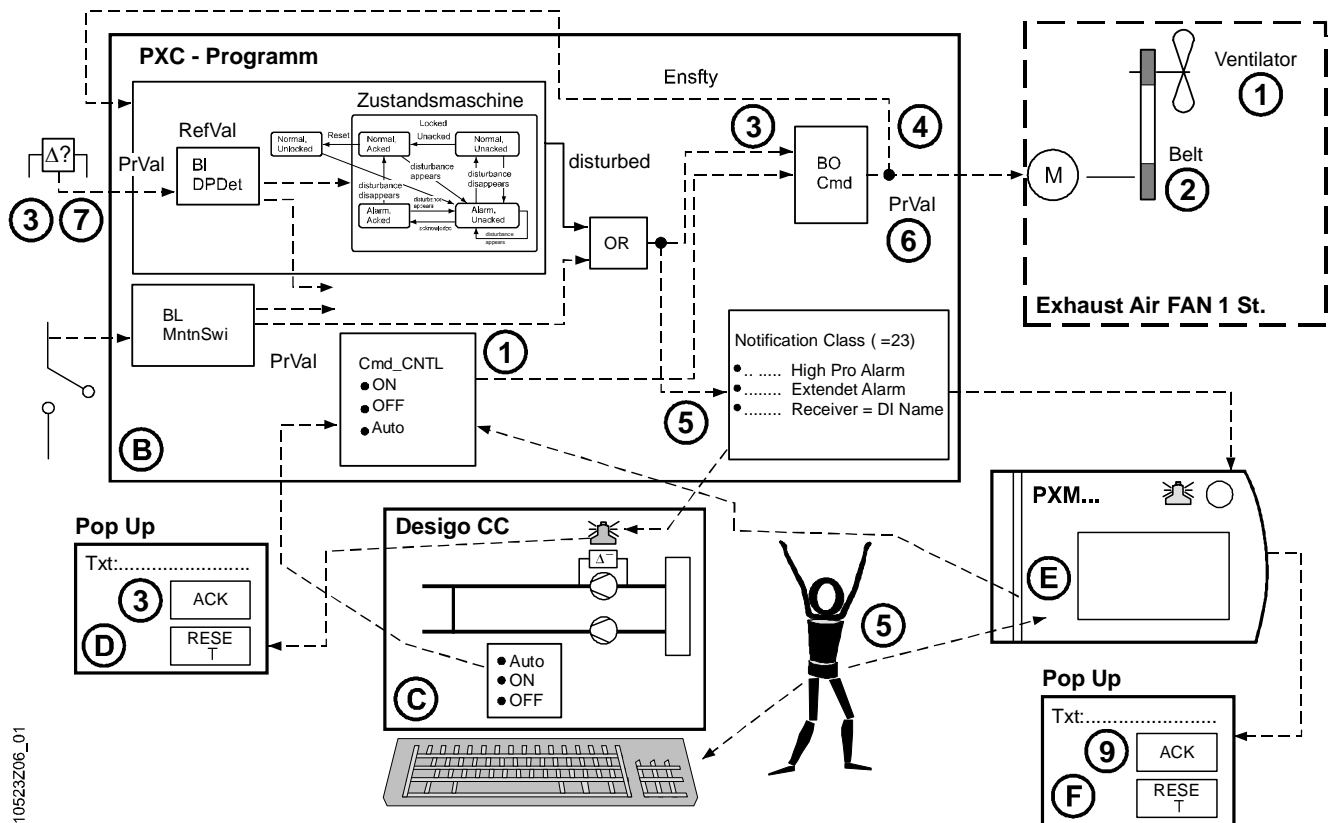
If, while in the OFFNORMAL state, a FAULT alarm condition occurs, there is then a state transition TO_FAULT (because as stated above, FAULT takes priority over OFFNORMAL).

9.2 Alarm example

What happens in the Desigo system when the V-belt of an extractor fan breaks?

The following figure shows the main information exchanged by the elements concerned, namely:

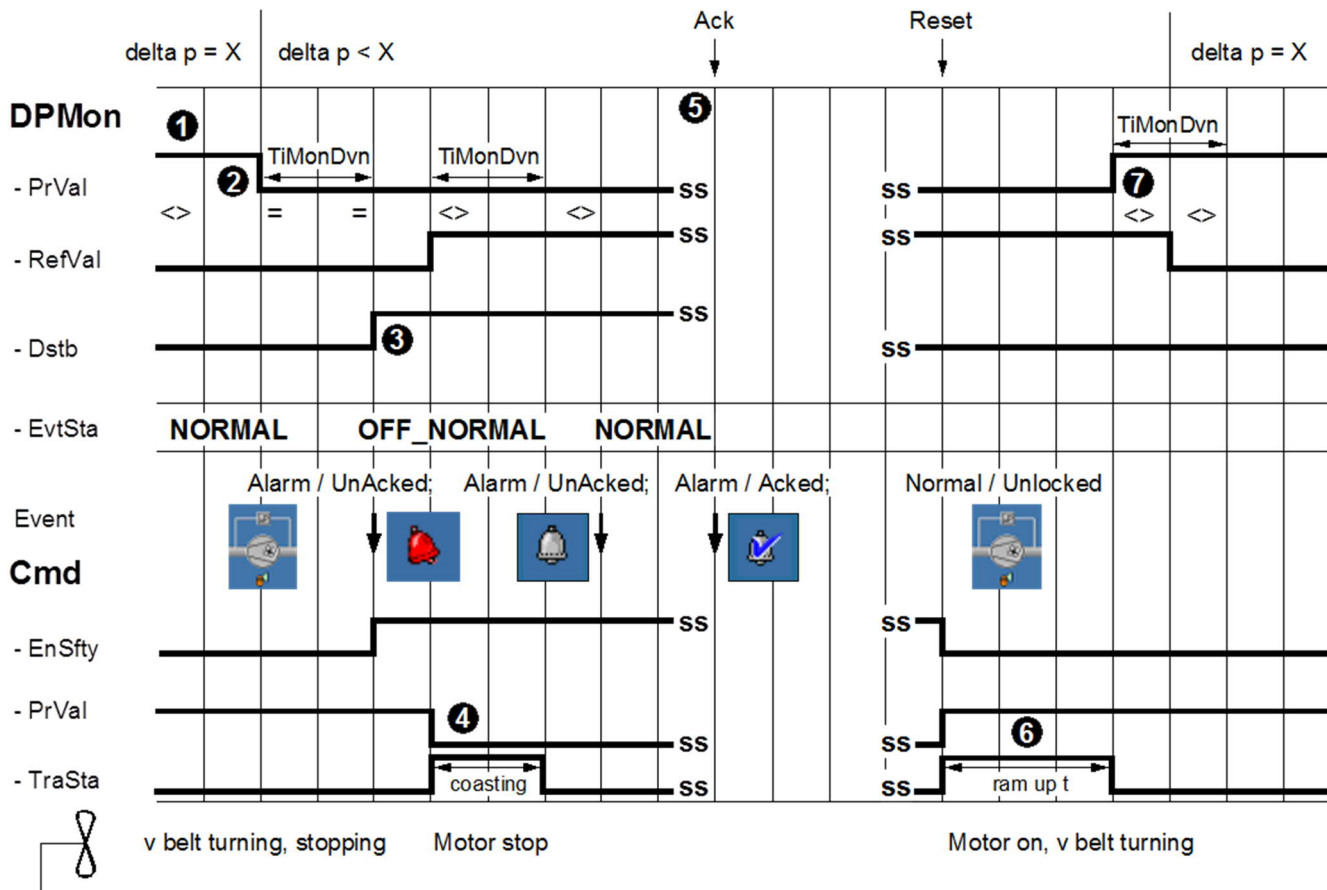
- Operator
- Ventilation system sensor/actuator (differential pressure monitor, maintenance switch and single-speed extract air fan)
- PXC program
- Desigo CC plant graphic
- PXM... operator unit



Key

10523Z06_01

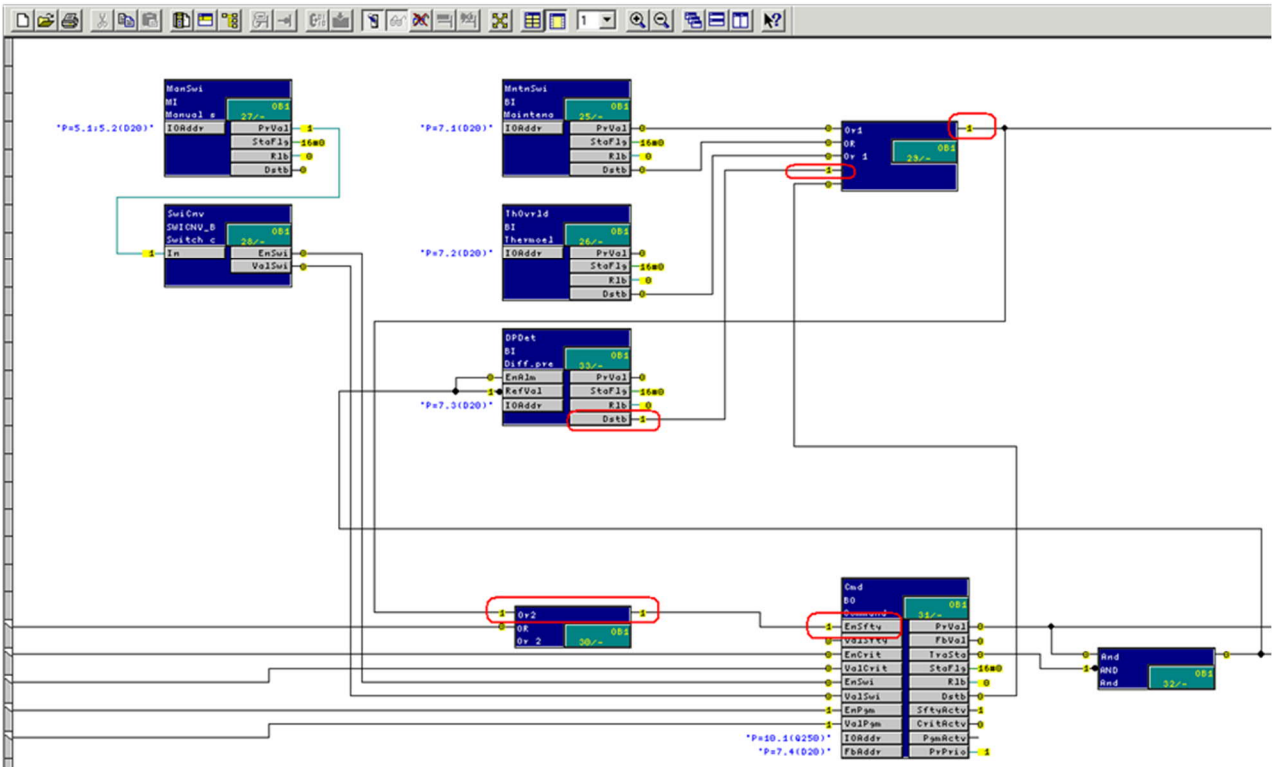
- State machine
- CFC program
- Desigo CC plant graphic page
- Desigo CC popup
- PXM... Values (in a PXM10 alarm handling is only possible for connected PXC's or PXR's)
- PXM... Popup (in a PXM10 alarm handling is only possible for connected PXC's or PXR's)



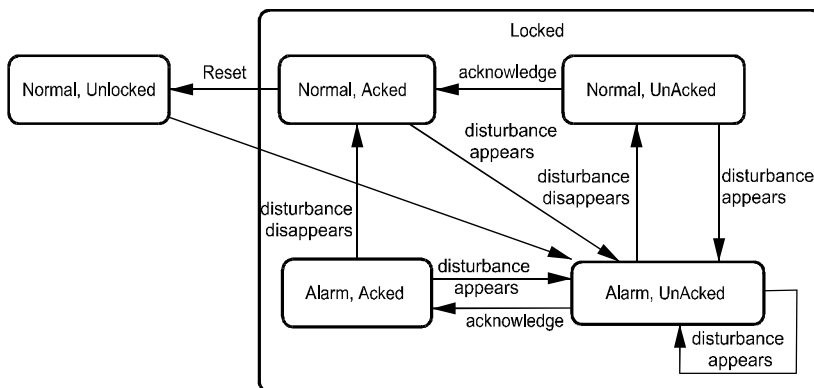
Time sequence in the example:

1. Ventilation system on (e.g., in automatic mode, $\text{Cmd.ValPgm} = 1$), single-speed extract air fan running, fan blades rotating
2. The V-belt breaks, the pressure drops, the differential pressure monitor responds ($\Delta p < X$) and DPMon.PrVal goes to zero. This activates the alarm monitoring function in the DP monitoring block, and the $[\text{TiMonDvn}]$ timer starts counting down.
3. After expiry of the time $[\text{TiMonDvn}]$, the DPMon block (BI) establishes that $\text{DPMon.PrVal} (0)$ is still equal to $\text{DPMon.RefVal} (0)$. This is equivalent to the OFFNORMAL state. DPMon.Dstb then goes to 1, and a TO_OFFNORMAL event is transmitted.
An alarm pop-up window is then displayed, in which the alarm message reads Alarm, Unacked.
4. The motor of the single-speed extract air fan is disabled (that is, $\text{Cmd.PrVal} \rightarrow 0$) because $[\text{EnSfty} \rightarrow 1$ and $\text{Cmd.ValSfty}=0$, Prio1 Cmd Input]. As a result, DPMon.RefVal goes to 1, thereby activating the alarm monitoring function. After expiry of the time $[\text{TiMonDvn}]$, the alarm monitoring function determines that $[\text{DPMon.PrVal} (0) \neq \text{DPMon.RefVal} (1)]$. The state therefore changes to NORMAL and a TO_NORMAL event is transmitted.
The alarm display now changes to Alarm = Normal, UnAcked.
5. The operator now acknowledges the alarm with Ack in the alarm pop-up dialog box. The alarm display now changes to Alarm = Normal, Acked. The operator sets the maintenance switch $[\text{MntnSwi}]$ to Maintenance ON, replaces the fan belt, returns the maintenance switch to Maintenance OFF and resets the alarm with Reset.
The alarm in the display changes to Alarm = Normal, Unlocked and $\text{DPMon.Dstb} \rightarrow 0$.

6. The fault has been cleared. When $DPMon.Dstb = 0$, then $Cmd.EnSfty \rightarrow 0$ and hence $Cmd.PrVal \rightarrow Cmd.ValPgm=1$, that is, the fan motor is enabled. Then, with $Cmd.TraSta = 1$ (transient state), the fan ramp-up time is allowed to expire, that is, $DPMon.RefVal$ is held at 1 during the transient state. Only after expiry of the ramp up time does $DPMon.RefVal$ revert to 0.
7. The ventilation system is already running (from step 6 on), that is, the fan blades start rotating, the pressure builds up and the differential pressure monitor detects $\Delta p = X$ again, that is, $DPMon.PrVal \rightarrow 1$. The alarm monitoring function is active again. After expiry of the time $[TiMonDvn]$, this determines that there is no alarm condition present, because $[DPMon.PrVal(0) \neq DPMon.RefVal(1)]$. The system then operates 100% correctly as described under step 1 above.



To simplify the time chart shown above, the connection to $DPMon.EnAlm$ has not been included.



9.3 Effects of BACnet properties on alarm response

BACnet properties in function blocks

- I = Input
- O = Output
- V = Value

| SBT designations | | | BACnet property | Function blocks (BACnet objects) | | | |
|-----------------------------|---------------|-----------------------------|------------------------|---|--------|--------|------------|
| Long name | Ref. | Description | | Other | Binary | Analog | Multistate |
| Alarm enable | EnAlm | Alarm enable | Alarm_Enable | Pulse Converter Event Enrollment | I/O/V | I/O/V | I/O/V |
| Event enable | EnEvt | Event enable | Event_Enable | Pulse Converter Command Control ¹ Power Control ¹ AlarmCollection Trend Logs Event Enrollment Loop | I/O/V | I/O/V | I/O/V |
| Event detection enable | EnEvtDet | Enable event detection | Event_Detection_Enable | Event Enrollment | | | |
| Event state | EvtSta | Event state | Event_State | Discipline I/O ¹ Group ¹ Pulse Converter Trend Logs Device-Info ¹ Command Control ¹ Power Control ¹ AlarmCollection Event Enrollment Loop | I/O/V | I/O/V | I/O/V |
| Feedback value | FbVal | Feedback value | Feedback_Value | | O | O | O |
| Upper limit | HiLm | Hi Limit | High_Limit | Pulse Converter | | I/O/V | |
| Limit enable | EnLm | Enable limit | Limit_Enable | | | | |
| Lower limit | LoLm | Low limits | Low_Limit | Pulse Converter | | I/O/V | |
| Message text | MsgTxt/EvtMsg | Message text | Message_Text | Discipline I/O ¹ Group ¹ Pulse Converter Command Control ¹ Power Control ¹ Loop Event Enrollment | I/O/V | I/O/V | I/O/V |
| Deviation monitoring period | TiMonDvn | Deviation monitoring period | Time_Delay | Pulse Converter Power Control ¹ Loop | I/O/V | I/O/V | I/O/V |
| Switch-off monitoring time | TiMonOff | Switch-off monitoring time | Time_Delay2 | | I/O/V | | |
| Switch-on monitoring time | TiMonOn | Switch-on monitoring time | Time_Delay1 | | I/O/V | | |
| Dead band or dead zone | Nz | Neutral zone | Deadband | Pulse Converter | | I/O/V | |
| Out of service | OoServ | Out of order | Out_of_Service | Device-Info ¹ Discipline I/O ¹ Group ¹ Pulse Converter Command Control ¹ Power Control ¹ AlarmCollection Event Enrollment Loop | I/O/V | I/O/V | I/O/V |

| SBT designations | | | BACnet property | Function blocks (BACnet objects) | | | |
|--------------------------------|-----------|---|--------------------------------|---|--------|--------|------------|
| Long name | Ref. | Description | | Other | Binary | Analog | Multistate |
| Present value | PrVal | Present value | Present_Value | Pulse Converter Command Control ¹ Power Control ¹ AlarmCollection Loop | I/O/V | I/O/V | I/O/V |
| Reference value | RefVal | Reference value | Alarm_Value | | I/V | | |
| Reference values | RefVals | Reference values | Alarm_Values | | | | I/V |
| Reliability | Rlb | Reliability | Reliability | Device-Info Discipline I/O ¹ Group ¹ Pulse Converter Trend Logs Command Control ¹ Power Control ¹ AlarmCollection Event Enrollment Loop | I/O/V | I/O/V | I/O/V |
| State flag | StaFlg | State flag | Status_Flags | Device-Info Pulse Converter Command Control ¹ Power Control ¹ AlarmCollection Event Enrollment Loop | I/O/V | I/O/V | I/O/V |
| Suppress event algorithm | SupEvtDet | Event algorithm inhibit | Event_Algorithm_Inhibit | Event Enrollment | | | |
| Event time stamp | TiStmEvt | Event time stamp | Event_Time_Stamp | Device-Info ¹ Discipline I/O ¹ Group ¹ Pulse Converter Trend Logs Command Control ¹ Power Control ¹ AlarmCollection Event Enrollment Loop | I/O/V | I/O/V | I/O/V |
| Notification function selector | NotifSel | Notification function selector [NotifSel] | Notification_Function_Selector | Device-Info ¹ Discipline I/O ¹ Group ¹ Pulse Converter Trend Logs Command Control ¹ Power Control ¹ AlarmCollection Event Enrollment Loop | I/O/V | I/O/V | I/O/V |

Key

¹ Only in Designo PX.

Alarm enable [EnAlm]

[EnAlm] (Boolean type) is used to enable and disable the monitoring of OFFNORMAL alarms. OFFNORMAL alarms will only be detected if [EnAlm] is TRUE. This is equivalent to the standard BACnet property Alarm_Enable.

FAULT alarms are monitored independently of the value of the alarm enable property [EnAlm]. Monitoring is continuous and cannot be disabled.

If [EnAlm] is changed from TRUE to FALSE during operation, the timer for all deviation monitoring periods [TiMonDvn] will be reset to zero. As soon as the value of [EnAlm] reverts to TRUE, the associated [TiMonDvn] timer starts counting to its preset value again from zero.

The value of [EnAlm] can be modified via BACnet clients or using the CFC editor online. During operation, if [EnAlm] is changed from TRUE to FALSE while an OFFNORMAL alarm is still active, this will result in an immediate state transition to TO_NORMAL1. In other words, the existing OFFNORMAL alarm condition is seen as having cleared, and the alarm state of the alarm source is updated accordingly.

Enable event [EnEvt]

[EnEvt] (Boolean type) is used to enable and disable the transfer of OFFNORMAL and FAULT alarms. OFFNORMAL and FAULT alarms are only transferred if [EnEvt] is TRUE. This is equivalent to the standard BACnet property Event_Enable.

Enable event detection [EnEvtDet]

[EnEvtDet] (Boolean type) lets you turn the intrinsic/algorithmic reporting on/off. OFFNORMAL and FAULT alarms are only forwarded when [EnEvtDet] = TRUE. This is equivalent to the standard BACnet property Event_Detection_Enable.

Event state [EvtSta]

This variable denotes the current alarm state of the object. It can accept three values: NORMAL, OFFNORMAL (in the case of analog HIGH_LIMIT and LOW_LIMIT values) and FAULT. The value of the variables is updated immediately after the associated alarm state transition. This is equivalent to the standard BACnet property Event_State.

Feedback value [FbVal]

[FbVal] is a feedback signal input, configured at a physical input via a separate hardware address. This use of a physical input can also be the source of reliability errors. [FbVal] can be neither overridden nor commanded. If [FbVal] is not configured at a physical input, then, by definition, it will be equal in value to Present Value, in which case no OFFNORMAL alarms can be issued via the output object. This is equivalent to the standard BACnet property Feedback_Value.

Unlike the binary output and multistate output blocks, the analog output function block does not use [FbVal] as a criterion for OFFNORMAL alarm conditions. If [FbVal] is used, it can be a source of reliability errors and can result in FAULT alarms.

Hi limit [HiLm]

This parameter (data type Real) determines the high alarm limit. If [PrVal] exceeds the high limit value [HiLm] for longer than the period defined under [TiMonDvn], an OFFNORMAL alarm condition prevails, namely: HIGH_LIMIT.

Enable limit [EnLm]

This variable only exists in the BACnet view of analog blocks (for reasons of compatibility with the BACnet standard). It has exactly the same meaning as the alarm enable variable [EnAlm] and its current value is derived from the value of [EnAlm] (that is, [EnLm = EnAlm], Limit enable = Enable alarm). This variable is equivalent to the standard BACnet property Limit_Enable.

Low limit [LoLm]

This parameter (data type Real) defines the low alarm limit. If [PrVal] exceeds the high limit value [LoLm] for longer than the period defined under [TiMonDvn], an OFFNORMAL alarm condition prevails, namely: LOW_LIMIT. This is equivalent to the standard BACnet property Low_Limit.

Message text [MsgTxt]

For Desigo PX, the variables [MsgTxt] or [EvtMsg] contain the message text of the last event notification associated with TO_OFFNORMAL, TO_FAULT and TO_NORMAL alarms.

Deviation monitoring period [TiMonDvn]

This refers to a delay before generating the alarm if an alarm condition is detected without a prior change in switch command (that is, without a set point change). [TiMonDvn] is not an integrating function, that is, the condition causing a change in the alarm state must persist without interruption for a period of time at least equivalent to the duration of [TiMonDvn], before it has any effect. The BACnet standard only supports a [TiMonDvn] for a monitoring period and the associated alarm delay. This is equivalent to the standard BACnet property Time_Delay.

In certain applications, different end-switch monitoring periods are required for Open and Close commands and for the Idle state.

For this reason, the additional properties [TiMonOff] and [TiMonOn] have been introduced for the binary input, binary output, binary value and multistate output objects.

Switch off- [TiMonOff] and switch on monitoring period [TiMonOn]

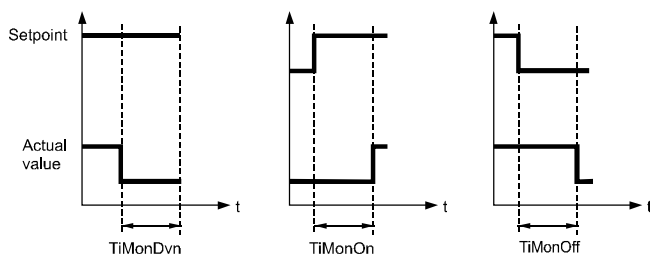
[TiMonOff]

Delay time before an alarm is generated when there is a preceding set point enable command. This is equivalent to proprietary BACnet properties Time_Delay1 and Time_Delay2.

[TiMonOn]

Delay time before an alarm is generated in the event of a set point switch-off command.

Application: Control of fire protection dampers (see further below).



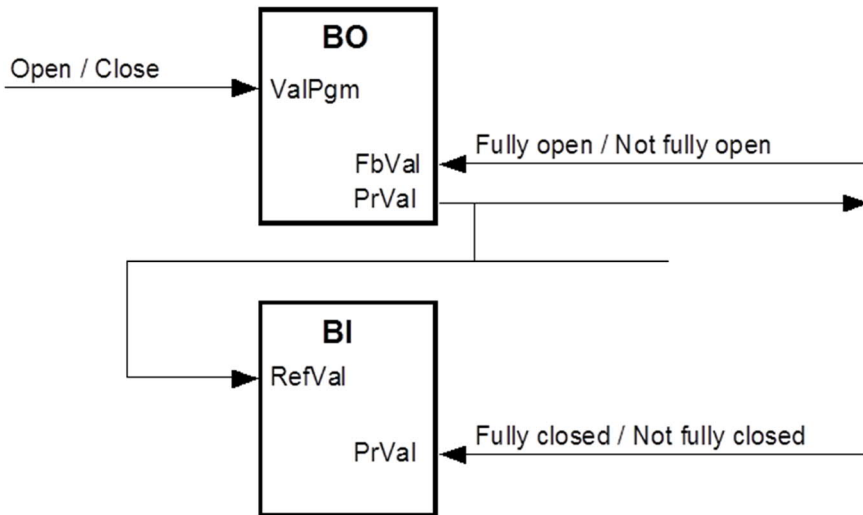
The definitions of the set point and the measured value depend on the object type:

| Object type | Set point | Measured value |
|---------------|-----------------|----------------|
| Binary Input | invers [RefVal] | [PrVal] |
| Binary Output | [PrVal] | [FbVal] |
| Binary Value | invers [RefVal] | [PrVal] |

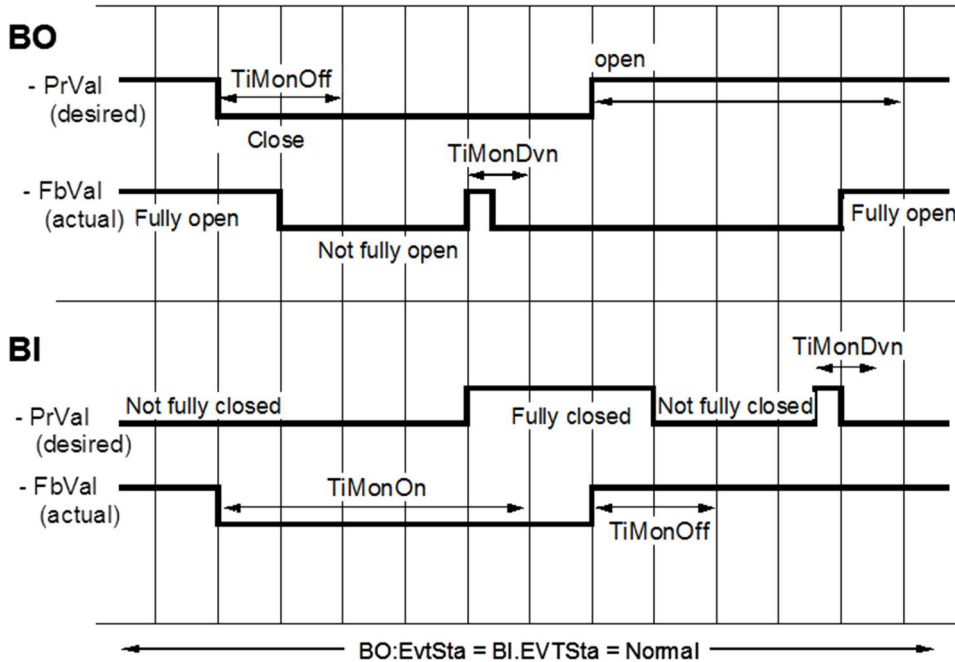
Examples

The following example shows the use of the three time periods [TiMonDvn], [TiMonOn], [TiMonOff]. For another example, see Alarm Example.

It is assumed that a fire damper has two separate feedback mechanisms (end switches). This means that the damper is commanded via the commands Open and Close. The first end switch, the Open switch delivers the signals Fully open or Not fully open. The second end switch, the Closed switch delivers the signals Fully closed or Not fully closed. The following is an example of how to connect the BO (binary output for commanding and integrating the Open switch) and BI (the binary output for the closed switch):

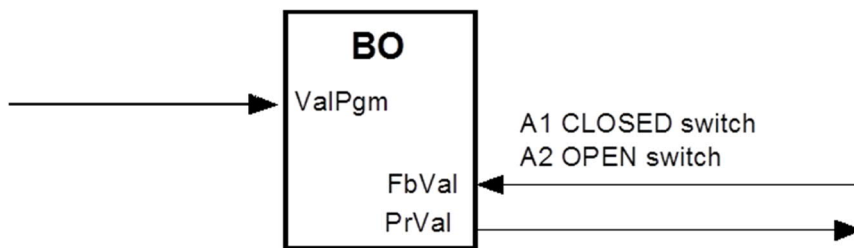


Given the feedback signal [FbVal] Fully open, the Open and Close commands follow the time sequence shown below, making use of all three deviation monitoring times [TiMonDvn].

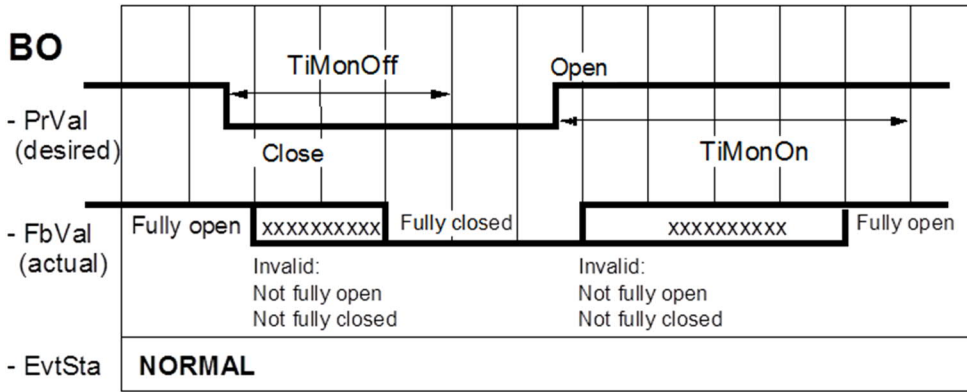


Since the BO block can handle the feedback of two different addresses, the fire-protection damper solution can be further simplified by direct connection of the Closed switch (Addr. 1) and Open switch (Addr. 2). In cases where both end switches are simultaneously On or simultaneously Off, the BO block treats the [FbVal] as invalid. Throughout this period, therefore, the alarm monitoring function will return the value Alarm = OFFNORMAL. The circuit and time sequence for normal and error conditions are as follows:

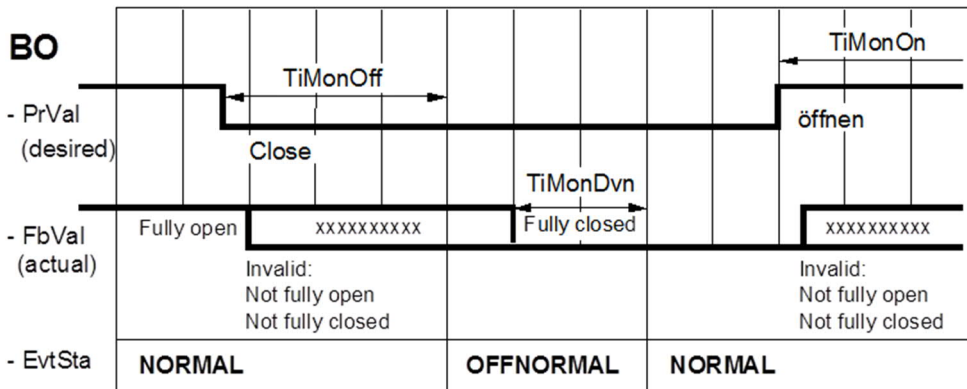
Circuit and time sequence for normal and error conditions:



Fire protection damper timing with BO and two feedback addresses:



Fire protection damper timing with BO and two feedback addresses:



Error condition: The damper does not close quickly enough.

Neutral zone [Nz]

[Nz] (data type Real) can be used to define a switching hysteresis for the state transition TO_NORMAL1. This is equivalent to the standard BACnet property Deadband.

Out of service [OoSrv]

The following applies to alarm response:

[PrVal] can also change for [OoSrv=TRUE].

[PrVal] is monitored for alarms irrespective of the source of any change in [PrVal]. In other words, the value of [OoSrv] does not affect the monitoring of OFFNORMAL alarms. If [OoSrv = TRUE], the [Rib] property can be overwritten via BACnet. However, the alarm monitoring system responds to changes in Reliability in the same way as if [OoSrv=FALSE]. This makes it possible to simulate FAULT alarms.

In the BACnet Device Info object, this Boolean variable is FALSE at the very time when the operating state is RUN, that is, when the D-MAP program is being run on the automation station. All the alarm-generating blocks (including the BACnet Device Info Object) are only monitored in the operational status RUN. Corresponds to the standard BACnet Property Out_of_Service.

Present value [PrVal]

OFFNORMAL alarms are monitored exclusively on the basis of the current value of [PrVal] the present value variable. The source of this present value (whether a process value, operator value, replacement value or commanded value) is irrelevant. This is equivalent to the standard BACnet property Present_Value.

Reliability [Rib]

The value under [PrVal] is only plausible if [Rib] = NO_FAULT_DETECTED.

When [Rib] <> NO_FAULT_DETECTED, this is precisely the condition for a FAULT alarm.

The BACnet Device Info Object is an exception. The value of [Rib] for the BACnet device object is NO_FAULT_DETECTED, except in the case of the fault FLASH_FULL (FAULT condition). This is equivalent to the standard BACnet property Reliability.

Reference value [RefVal]

[RefVal] is a set point, used to set the value which [PrVal] (the measured value) must assume in order to initiate an alarm after the time defined by [TiMonDvn] has expired. [RefVal] is equivalent to the standard BACnet property Alarm_Value.

Reference values [RefVals]

The variable [RefVals] comprises a list of multistate elements. The value range (number of states) of the items in the list is the same as for [PrVal]. All states to be treated as OFFNORMAL are entered under [RefVals]. [RefVals] is equivalent to the standard BACnet property Alarm_Values.

Example of [RefVals] : STEP 1, STEP 2, STEP 4

| Name | Value |
|---------|--------|
| State 1 | STEP 1 |
| State 2 | STEP 2 |
| State 3 | STEP 4 |

In this example, an incoming OFFNORMAL alarm will be detected if [PrVal] = STEP 1, STEP 2 or STEP 4 after expiry of the period defined by [TiMonDvn].

State flag [StaFlg]

The variable [StaFlg] includes the two bits 'In_Alarm' and 'Fault'.

By definition, In_Alarm is TRUE whenever [EvtSta] is not equal to NORMAL.

By definition, FAULT is TRUE whenever [EvtSta = FAULT].

The value of these two [StaFlg] variables is thus derived from another variable.

For each change of the variable [StaFlg] a Change of Value (COV) notification is sent to all COV subscribers of the alarm-generating object. In this way, the COV subscribers can be kept informed of an alarm state in their COV server. This is equivalent to the standard BACnet property Status_Flags.

Suppress event detection [SupEvtDet]

[SupEvtDet] (Boolean type) lets you turn the OFFNORMAL and FAULT alarm detection on/off. OFFNORMAL and FAULT alarms are only detected when [SupEvtDet] = FALSE. This is equivalent to the standard BACnet property Event_Algorithm_Inhibit.

Event time stamp [TiStmEvt]

This variable (ARRAY [3], type TimeStamp), contains time stamps for the last changes of state of the alarm-generating object TO_OFFNORMAL, TO_FAULT and TO_NORMAL. The value of the variables is updated immediately after the associated alarm state transition. This is equivalent to the standard BACnet property Event_Time_Stamps.

Notification function selector [NotifSel]

This variable specifies if the alarm function is executed as per default pattern (Simple-/Basic-/Extended alarm) or as per a customized alarm function.

9.4 Alarm response of the function blocks

Alarm Collection

The default value of [EnEvt] for the Alarm Collection object is FALSE, that is, [EvtSta] transitions are not notified.

An OFFNORMAL alarm is generated when:

- The following applies to one or more alarm collection members:
[EvtSta] <> NORMAL and applies simultaneously for all these members: [StaFlg].Fault = false.

A FAULT alarm is generated when:

- The following applies to one or more alarm collection members:
[StaFlg].Fault = true and therefore is set [Rib] = UNRELIABLE_MEMBERS.

Analog Input, Analog Value, Analog Output

The Analog Input, Analog Value and Analog Output function blocks all have an identical alarm handling procedure.

The analog output function block also has a feedback value [FbVal]; however, this is not used for alarm monitoring. High and low alarm limits (variables [HiLm] and [LoLm]) are set for the OFFNORMAL alarms of analog objects. An OFFNORMAL alarm occurs either when the high alarm limit is exceeded, or when the current value falls below the low alarm limit. OFFNORMAL alarms are thus subdivided into two subcategories: HIGH_LIMIT and LOW_LIMIT. In addition, the variable [Nz] can be used to define a switching hysteresis for [HiLm] and [LoLm] to prevent over-frequent switching of alarms around the alarm limit.

Alarm response

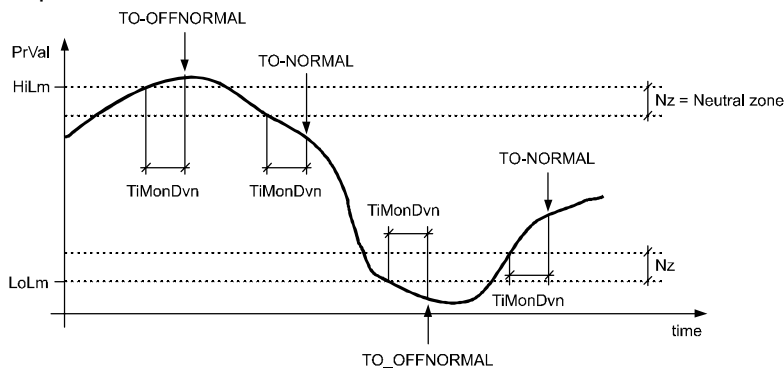
An OFFNORMAL alarm is generated:

- [PrVal] has either remained above the high alarm limit specified by the [HiLm] variable for a period of time longer than the period specified in [TiMonDvn]
- or [PrVal] has remained below the low alarm limit specified by the [LoLm] for a period of time longer than the period specified in [TiMonDvn]

An existing OFFNORMAL (HIGH_LIMIT) alarm will disappear when [PrVal] has remained below the value $([HiLm] + [Nz])$ for longer than the time specified in the variable [TiMonDvn].

An existing OFFNORMAL (LOW_LIMIT) alarm will disappear when [PrVal] has remained below the value $([HiLm] + [Nz])$ for longer than the time specified in the variable [TiMonDvn].

- A FAULT alarm is generated as soon as the [Rlb] property of the function block assumes any value other than NO_FAULT_DETECTED. In particular, this is the case when [Rlb] changes from a value not equal to NO_FAULT_DETECTED to another value not equal to NO_FAULT_DETECTED.
- A FAULT alarm will disappear as soon as the [Rlb] property of the function block changes from a value not equal to NO_FAULT_DETECTED back to the value NO_FAULT_DETECTED.



Event state: NORMAL | HIGH_LIMIT | NORMAL | LOW_LIMIT | NORMAL

BACnet Device Info Object

OFFNORMAL alarms

All the alarm-generating objects described so far model specific types of individual data points (physical or virtual). The BACnet device object by contrast, models the properties of an automation station as a complete entity. Alarm-relevant faults which cannot be allocated to a data point can be generated in an automation station (see the examples further below). This is why the BACnet device object includes an alarm mechanism. The alarm state machine and the alarm-related variables are essentially the same as for all the other alarm-generating block types. The difference lies in the possible causes of the alarm:

The alarm conditions described below cause the generation of an OFFNORMAL alarm in the BACnet Device Object:

Battery low

The battery in an automation station is checked periodically. An alarm is generated if the battery voltage is too low, or if the battery itself is missing. When the required voltage level is reached again, the alarm is reset with BATTERY_NOT_LOW.

RAM Pattern failed

This indicates that a memory-check error was found when the automation station was switched on. If no memory-check error is detected when the automation station is next switched on, the alarm will be reset.

Recipient not receivable

A recipient name (e.g., the configured recipient of an alarm) could not be resolved, because, e.g., the network connection to the recipient was interrupted. This causes an alarm to be generated. The alarm is cleared as soon as the subsequent name resolution process succeeds.

Notif. Class ref. missing

Each alarm-generating block includes a reference to a Notification Class block. If the referenced Notification Class block does not exist, the BACnet Device Object generates an alarm.

Life check error

While the life check is in progress, the primary server finds that it is unable to communicate with one or more of its backup servers (e.g., owing to a network failure). This causes an alarm to be generated. The alarm is cleared when, during a subsequent life check, all the backup servers are found again.

Primary server not found

This bit is set when the backup server detects that the primary server is no longer connected to the network. At the same time a notification (data-type STRING) is sent, defining the source, target and reason. The bit is reset as soon as the backup server detects the primary server on the network again.

FAULT alarms

The condition described below causes a FAULT alarm to be generated in the BACnet Device Object:

Flash is full

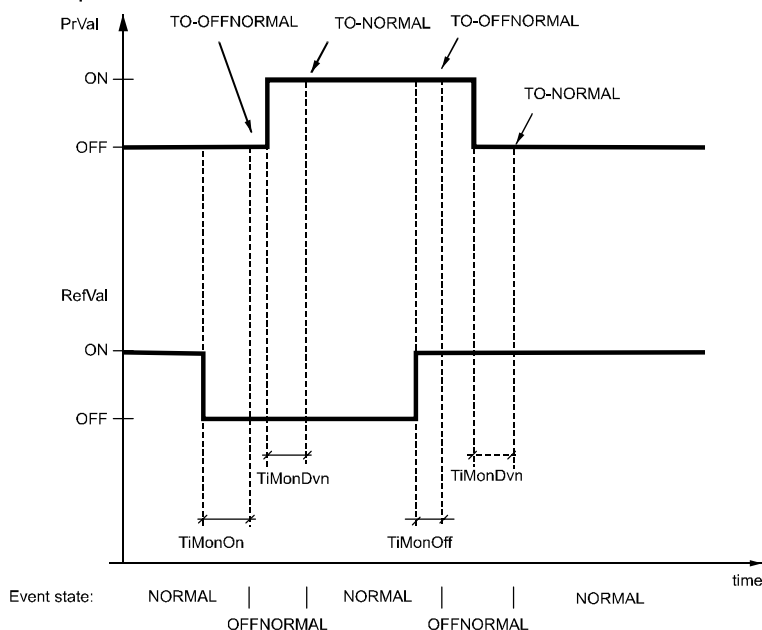
The automation station checks periodically whether there is at least one free page (64 kB) in the flash memory. This bit is set if the flash memory falls below this value. The bit is reset when the flash memory contains at least one free page again.

Alarm response of the BACnet Device Object is also parameterized or depicted by the number of variables, but the display differs: The BACnet Device Object is not displayed by a D-MAP function block, but rather only visible via BACnet. The variables described are therefore only accessible as properties of the BACnet Device Object.

Binary Input and Binary Value

The alarm handling process is identical for the function blocks Binary Input and Binary Value.

- An OFFNORMAL alarm occurs when [PrVal] assumes the value specified by the variable [RefVal] for a time period at least equivalent to the delay time specified in the variable [TiMonDvn], [TiMonOff] or [TiMonOn].
- An existing OFFNORMAL alarm condition will disappear (a) when [PrVal] assumes the value complementary to [RefVal] for a period at least equivalent to the period specified in [TiMonDvn], [TiMonOff] or [TiMonOn] or (b) when [EnAlm] is changed from TRUE to FALSE (see further below).
- A FAULT alarm is generated when the [Rlb] property of the function block assumes any value other than NO_FAULT_DETECTED. In particular, this is the case when [Rlb] changes from a value not equal to NO_FAULT_DETECTED to another value not equal to NO_FAULT_DETECTED.
- A FAULT alarm will disappear as soon as the [Rlb] property of the function block changes from a value not equal to NO_FAULT_DETECTED back to the value NO_FAULT_DETECTED.



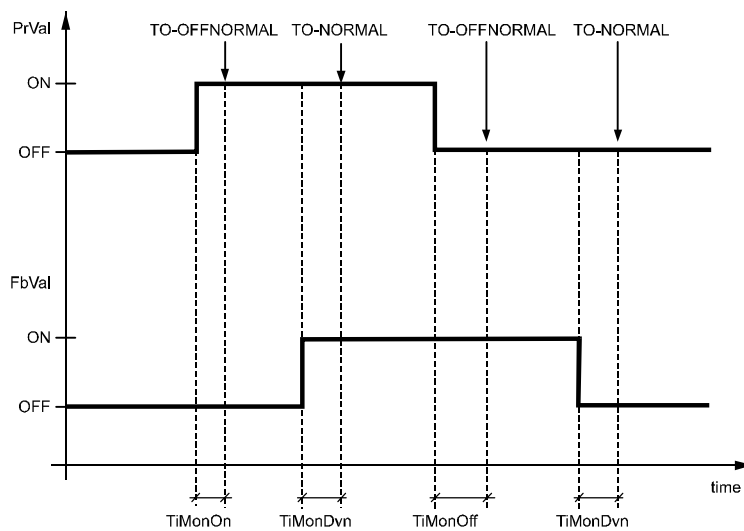
Binary Output

The alarm handling process in the binary output function block is essentially different from that of the binary input and binary value blocks.

- An OFFNORMAL alarm occurs when the current values of the variables [PrVal] and [FbVal] differ from each other for a time period at least equivalent to the delay time specified in [TiMonDvn], [TiMonOff] or [TiMonOn].
- An existing OFFNORMAL alarm will disappear when the current [PrVal] und [FbVal] are again identical and remain so for a period at least equivalent to the time specified in the variable [TiMonDvn].
- A FAULT alarm is generated when the [Rlb] property of the function block assumes any value other than NO_FAULT_DETECTED. In particular, this is the case when the [Rlb] property changes from a value not equal to NO_FAULT_DETECTED to another value not equal to NO_FAULT_DETECTED.

In the case of the binary output, [Rlb] errors may originate both from the [PrVal] (or associated physical output) and from the [FbVal] (or associated physical input).

- A FAULT alarm will disappear as soon as the variable [Rlb] changes from a value not equal to NO_FAULT_DETECTED back to the value NO_FAULT_DETECTED.



Event state: NORMAL | OFFNORMAL | NORMAL | OFFNORMAL | NORMAL

Command Control

An OFFNORMAL alarm is generated:

- A monitored, referenced object is not enabled
- A referenced object cannot be enabled

A FAULT alarm is generated when:

- A referenced object is not found
- A referenced object is not a commandable object (output object or value object)
- Invalid priorities are used for the referenced object (valid priorities are Priority 2, 5, 14 and 16)
- ProgramValue or ExceptionValue are outside the permissible range
- The referenced objects have a different number of operating modes
- The function table is empty

Discipline I/Os and Group

Alarm response

Alarm handling is identical for Discipline I/O and Group blocks. These function blocks only support FAULT alarms.

- A FAULT alarm is generated as soon as the [Rlb] property of the function block assumes any value other than NO_FAULT_DETECTED. In particular, this is the case when [Rlb] changes from a value not equal to NO_FAULT_DETECTED to another value not equal to NO_FAULT_DETECTED.
- A FAULT alarm will disappear as soon as the [Rlb] property of the function block changes from a value not equal to NO_FAULT_DETECTED back to the value NO_FAULT_DETECTED.

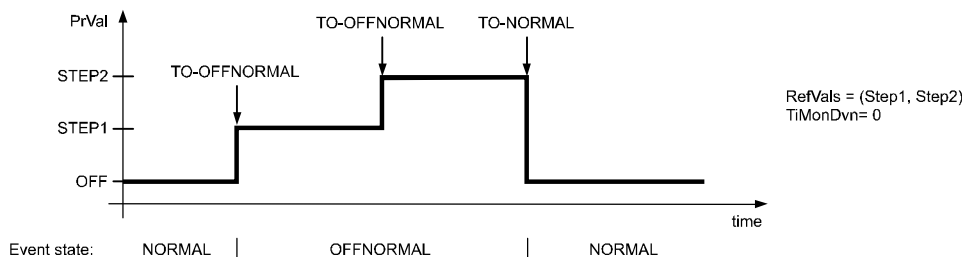
The following conditions cause a FAULT alarm to be initiated:

- **Address conflict:**
The subsystem fails to recognize the device defined in the [IOAddress] parameter. This alarm is issued by the associated function block.
- **Communications error:**
The subsystem indicates a communications failure. This can be due to a bus open circuit or a faulty device, or, very rarely, to a communications overload on the bus. These alarms are indicated by the shared function block.
The subsystem indicates an inadmissible response from a device e.g. in the case of faulty QAX... room unit. These alarms are indicated by the shared function block.

Multistate Input and Multistate Value

The alarm handling process is identical for the function blocks Multistate Input and Multistate Value.

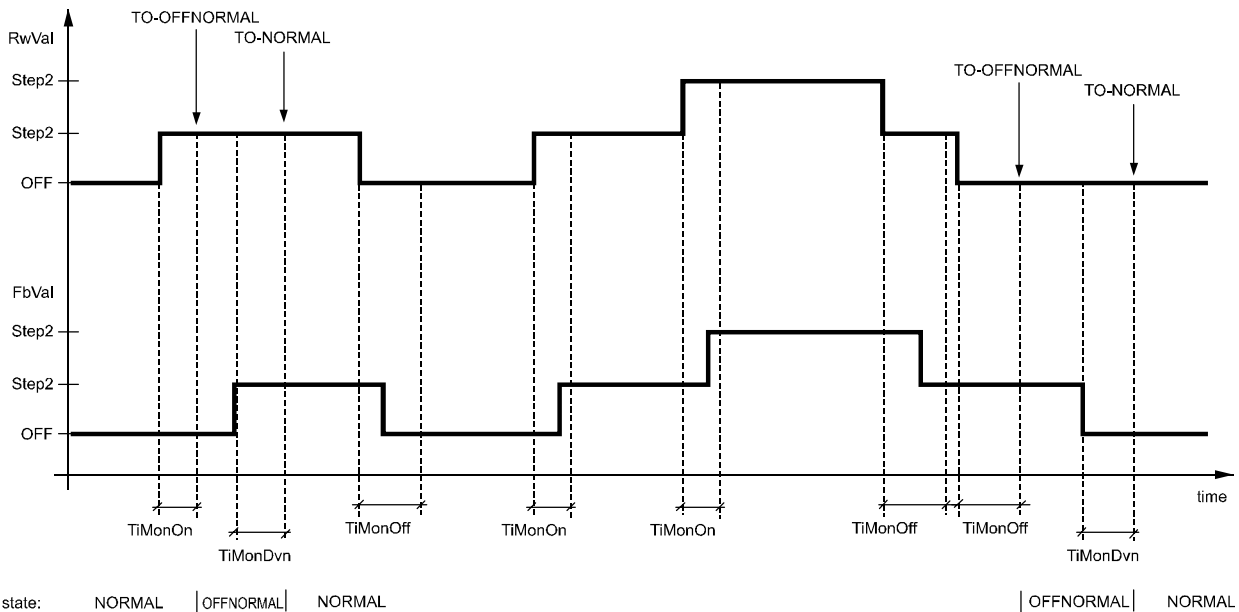
- An OFFNORMAL alarm occurs when [PrVal] assumes one of the values specified under [RefVals] (list of multistate values) and remains at this value for a period at least equivalent to the time specified by the variable [TiMonDvn]. In particular, this applies when [PrVal] changes from one value in [RefVals] to another value in [RefVals].
- An existing OFFNORMAL alarm condition will disappear either if [PrVal] reverts to a value not contained in the [RefVals] list, and retains this value for a period at least equivalent to the period specified in [TiMonDvn], or if [EnAlm] is changed from TRUE to FALSE (see further below).
- A FAULT alarm is generated when the [Rlb] property of the function block assumes any value other than NO_FAULT_DETECTED. In particular, this is the case when [Rlb] changes from a value not equal to NO_FAULT_DETECTED to another value not equal to NO_FAULT_DETECTED.
- A FAULT alarm will disappear as soon as the [Rlb] property of the function block changes from a value not equal to NO_FAULT_DETECTED back to the value NO_FAULT_DETECTED.



Multistate output

The alarm handling procedure for the Multistate Output function block is different from the alarm handling procedure for the Multistate Input and Multistate Value function blocks, but follows the same principles as for the Binary Output block:

- An OFFNORMAL alarm occurs when the current values of the variables [RwVal] and [FbVal] differ from each other for a time period at least equivalent to the delay time specified in [TiMonDvn].
- An existing OFFNORMAL alarm will disappear when the current [PrVal] und [FbVal] are again identical and remain so for a period at least equivalent to the time specified in the variable [TiMonDvn].
- A FAULT alarm is generated when the [Rlb] property of the function block assumes any value other than NO_FAULT_DETECTED. In particular, this is the case when the [Rlb] property changes from a value not equal to NO_FAULT_DETECTED to another value not equal to NO_FAULT_DETECTED. In the case of the multistate output block, [Rlb] errors may originate both from the [PrVal] (or associated physical output) and from [FbVal] (or associated physical input).
- A FAULT alarm will disappear as soon as [Rlb] changes from a value not equal to NO_FAULT_DETECTED back to the value NO_FAULT_DETECTED.



Power Control

An OFFNORMAL alarm is generated:

- The UP command is issued but the maximum stage has already been reached
- The UP command causes MaxPower to be exceeded
- Table_No is set outside the admissible range

A FAULT alarm is generated when:

- A referenced object is not found
- A referenced object is not a multistate value object
- Object_No. is outside the admissible range
- StepLimit is outside the range of the referenced object
- The function table is empty

Pulse Converter

Alarm response

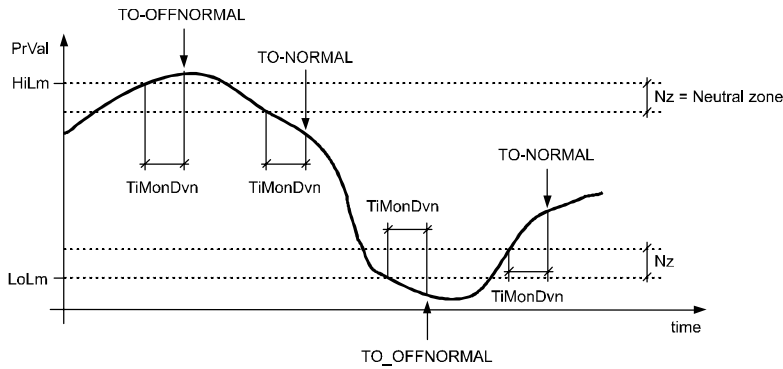
An OFFNORMAL alarm is generated, when [PrVal]:

- [PrVal] has remained above the high alarm limit specified by the [HiLm] variable for a period of time longer than the period specified in [TiMonDvn] (HIGH_LIMIT)
- or [PrVal] has remained below the low alarm limit specified by the [LoLm] variable for a period of time longer than the period specified in [TiMonDvn] (LOW_LIMIT)

An existing OFFNORMAL (HIGH_LIMIT) alarm will disappear when [PrVal] has remained below the value $([HiLm] + [Nz])$ for longer than the time specified in the variable [TiMonDvn]

An existing OFFNORMAL (LOW_LIMIT) alarm will disappear when [PrVal] has remained below the value $([HiLm] + [Nz])$ for longer than the time specified in the variable [TiMonDvn]

- A FAULT alarm is generated as soon as the [Rlb] property of the function block assumes any value other than NO_FAULT_DETECTED. In particular, this is the case when [Rlb] changes from a value not equal to NO_FAULT_DETECTED to another value not equal to NO_FAULT_DETECTED.
- A FAULT alarm will disappear as soon as the [Rlb] property of the function block changes from a value not equal to NO_FAULT_DETECTED back to the value NO_FAULT_DETECTED.



Event state: NORMAL | HIGH_LIMIT | NORMAL | LOW_LIMIT | NORMAL

Trend Log

Alarm response

The Trend Log function has an Intrinsic Reporting mechanism, but does not issue OFFNORMAL alarms.

- A FAULT alarm is generated as soon as the [Rib] property of the function block assumes any value other than NO_FAULT_DETECTED. In particular, this is the case when [Rib] changes from a value not equal to NO_FAULT_DETECTED to another value not equal to NO_FAULT_DETECTED.
- A FAULT alarm will disappear as soon as the [Rib] property of the function block changes from a value not equal to NO_FAULT_DETECTED back to the value NO_FAULT_DETECTED.

Event message

An event is generated when:

- The record count exceeds the record count value [RecCnt] set via the notification threshold [NotifThd], that is, the local non-volatile trend memory is overflowing.

Event Enrollment

The Event Enrollment object monitors referenced BACnet properties in other objects. The referenced property can be located in the local device or in another device.

Event algorithms

Monitoring details for a property value are defined by means of event algorithms. An event algorithm has a specific parameter. Event algorithms are the same as for Intrinsic Reporting. Intrinsic Reporting uses a subset of the possible event algorithms of Event Enrollment.

| Event_Type | Event_State | Event_Parameters | Data type |
|---------------------|-----------------------------------|---|---|
| CHANGE_OF_BITSTRING | NORMAL OFFNORMAL | Time_Delay Bitmask List_Of_Bitstring_Values | Unsigned BIT STRING list of BIT STRING |
| CHANGE_OF_STATE | NORMAL OFFNORMAL | Time_Delay List_Of_Values | Unsigned list of BACnetPropertyStates |
| CHANGE_OF_VALUE | NORMAL | Time_Delay Bitmask Referenced_Property_Increment | Unsigned BIT STRING choice {BIT STRING, REAL} |
| COMMAND_FAILURE | NORMAL OFFNORMAL | Time_Delay Feedback_Property_Reference | Unsigned BACnetDeviceObjectPropertyReference |
| FLOATING_LIMIT | NORMAL HIGH_LIMIT LOW_LIMIT | Time_Delay Setpoint_Reference Low_Diff_Limit High_Diff_Limit Deadband | Unsigned BACnetDeviceObjectPropertyReference REAL REAL REAL |
| OUT_OF_RANGE | NORMAL HIGH_LIMIT LOW_LIMIT | Time_Delay Low_Limit High_Limit | Unsigned REAL REAL |

| Event_Type | Event_State | Event_Parameters | Data type |
|-----------------------|--|---|---|
| | | Deadband | REAL |
| BUFFER_READY | NORMAL | Notification_Threshold Previous_Notification_Count | Unsigned Unsigned |
| CHANGE_OF_LIFE_SAFETY | NORMAL OFFNORMAL LIFE_SAFETY_ALARM | Time_Delay List_Of_Alarm_Values List_Of_Life_Safety_Alarm_Values Mode_Property_Reference | Unsigned list of BACnetLifeSafetyState list of BACnetLifeSafetyState BACnetDeviceObjectPropertyReference |
| EXTENDED | Any BACnetEventState | Vendor_Id Extended_Event_Type Parameters | Unsigned Unsigned Extended_Event_Type |
| UNSIGNED_RANGE | NORMAL HIGH_LIMIT LOW_LIMIT | Time_Delay Low_Limit High_Limit | Unsigned Unsigned Unsigned |

Event notification

An Event Enrollment object also monitors the status flag property of an object with referenced property. If the FAULT flag of the referenced object is set, the Event Enrollment object generates a Fault alarm.

Loop object

Alarm response

The Loop object contains intrinsic reporting.

An OFFNORMAL alarm occurs when:

- [XCtr] exceeds the limit (SetPoint + ErrorLimit) longer than the specified time (HIGH_LIMIT) defined in variable [TiMonDvn]
- [XCtr] drops below the limit (SetPoint – ErrorLimit) longer than the specified time (LOW_LIMIT) defined in variable [TiMonDvn]

An existing OFFNORMAL alarm (HIGH_LIMIT) disappears again when [XCtr] drops below the value (SetPoint + ErrorLimit – Deadband) longer than the specified time defined in variable [TiMonDvn].

An existing OFFNORMAL alarm (LOW_LIMIT) disappears again when [XCtr] exceeds the value (SetPoint – ErrorLimit + Deadband) longer than the specified time defined in variable [TiMonDvn].

FAULT alarm:

- A FAULT alarm occurs immediately as soon as [Rlb] of the function block has a value other than NO_FAULT_DETECTED. This is true in particular when [Rlb] changes from a value that is not equal to NO_FAULT_DETECTED to another value that is not equal to NO_FAULT_DETECTED.
- A FAULT alarm disappears immediately as soon as [Rlb] of the function block changes again from a value that is unequal to NO_FAULT_DETECTED to the value NO_FAULT_DETECTED.

9.5 Alarm functions

Depending on the type and degree of urgency of the alarm, the system user may be required to acknowledge a change in the alarm state with an explicit operator action.

Acknowledgement

There are two types of acknowledgement:

- Acknowledgement: Confirmation of an incoming alarm
- Reset: Confirmation that an alarm is no longer present

This type of interaction can be carried out locally or with clients, via the network.

Standard pattern

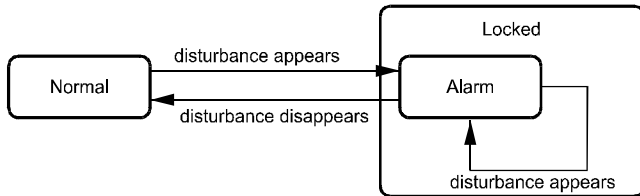
There are three standard categories of alarm, or alarm functions, reflecting the type of acknowledgement required:

- Simple alarm
- Basic alarm
- Extended alarm

Each alarm source is assigned (via a Notification Class, see further below) to one alarm function only. No further distinction is made at this stage between OFFNORMAL and FAULT alarms.

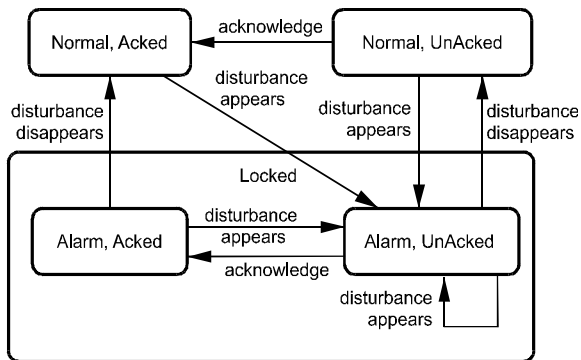
Simple alarm

Neither incoming alarms (disturbance appears) nor disappearing alarms (disturbance disappears) require acknowledgement.



Basic alarm

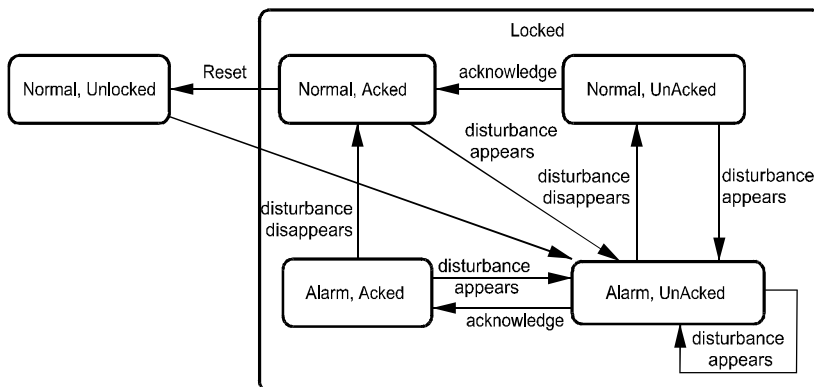
Acknowledgment is required for incoming alarms only, but not for alarms that have been cleared (that is, acknowledgement required but not reset).



Extended alarm

Locking alarm with acknowledgement of incoming alarms (disturbance appears) and cleared alarms (disturbance disappears). Alarms in this category require both acknowledgement and reset.

While testing the system, it may not be possible to reset an alarm. The reason is that an Extended Alarm is not reset until it has been acknowledged, and the time delay has expired.

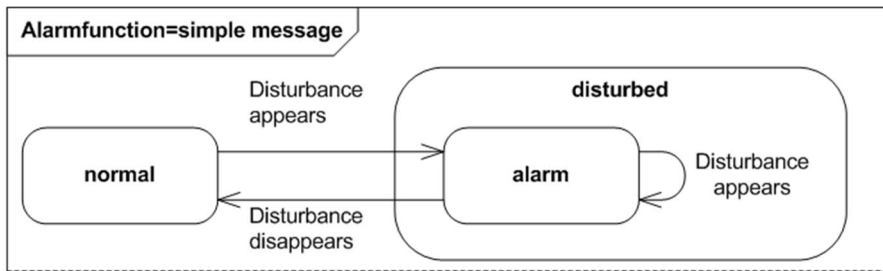


The alarm remains locked until the fault has disappeared and has been acknowledged and a reset has been carried out, e.g.:

The burner system is restarted when the service engineer has acknowledged the alarm, cleared the fault and reset the alarm. The alarm state of every alarm-generating object is managed within the object itself. The state machines above illustrate this for each of the alarm functions.

Simple message

The alarm function simple message is the same function as the simple alarm. State transitions, however, are not indicated as events, but alarms.



For HVAC applications and response in the system, the functionality is identical to simple alarm: Simple alarm without acknowledgement of incoming and outgoing faults. The only difference is EventNotification as alarm or event.

Customized alarm

Any alarm function under BACnet can be used. The following behavior can be defined for customized alarms:

- EventNotification can occur as either event or alarm
- Acknowledgement: For each change of state (TO-OFFNORMAL, TO-NORMAL, and TO-FAULT) can be defined whether or not an acknowledgement is required.

[AckTra] Acknowledged transitions

This feature is used to represent the acknowledgement status, or to handle information about which state transitions currently still require acknowledgement. The value of [AckTra] is based on the alarm function, the current [EvtSta] and the monitoring of acknowledgements already received.

[AckTra] consists of three flags, one each for TO-OFFNORMAL, TO-NORMAL and TO-FAULT. The flags have the following meanings:

- The flag is always FALSE when there has been a relevant state transition and an acknowledgement is required, because this is a requirement of the alarm function and no acknowledgement has yet taken place.
- The flag is TRUE when no acknowledgement of the state transition is required. This may be the case because the alarm function does not require acknowledgement, or because no state transition has occurred, or because a state transition that has occurred has already been acknowledged.

[TiAck] Time of acknowledgement

Time of the last acknowledgement (time stamp).

9.6 Alarm management by notification class

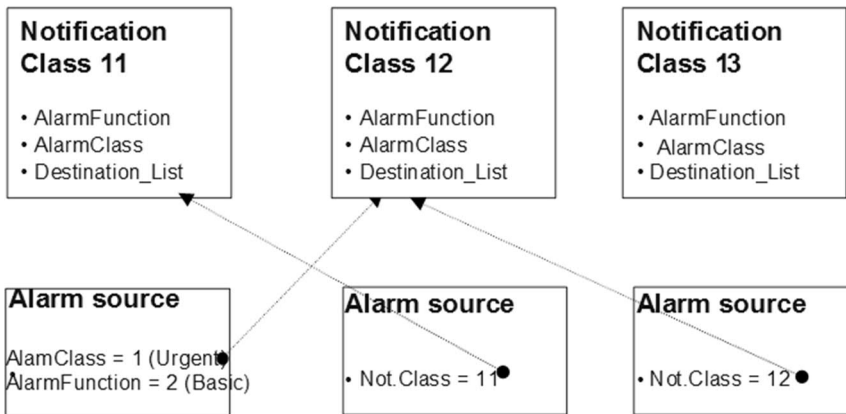
Intrinsic reporting

With intrinsic reporting, the alarmable object itself assumes alarm identification and state machine for alarm handling. However, the subsequent distribution of alarm messages to alarm clients and the alarm management is no longer the responsibility of the alarm source itself, but of a Notification Class object assigned to the alarm source. The Notification Class object is both a D-MAP function block and a standard BACnet object, which contains all the information required for the distribution of alarms and system events within the system.

Algorithmic reporting

With algorithmic reporting, alarm detection and the state machine for alarm handling normally are taken over by the Event Enrollment object. In this case, alarm management also is set up in an alarm source via assigned Notification Class object.

Notification Class



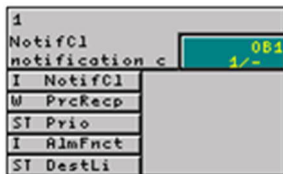
Each alarm-generating object is assigned one notification class [NotifCl] only, but one notification class can be used by more than one alarm-generating object. This makes it possible to create a Notification Class object for each group of alarms (e.g., HVAC alarms, fire alarms etc.). Each alarm source in a given alarm group is assigned to the [NotifCl] for that group.

There are global and local notification class objects:

- Global notification class: One set of max. 18 global notification class objects per site. Global notification classes are replicated and thus exist on all Desigo PX of a site in identical form.
- Local notification class: On Desigo PX, local notification classes can be engineered, but are NOT replicated.
- Desigo room automation supports exclusively local notification classes.

Interface definition

The notification class function block [NotifCl] is the means by which functionality is transferred from the BACnet standard into the CFC environment.



This function block contains the instance number of the Notification Class (an integer), which must be identical to the value entered in the subordinate alarm sources. This makes it possible to create a unique reference.

The number must not be modified online.

Notification class number

There are 18 predefined global notification classes. The notification class is identified with the two independent variables AlarmFunction and AlarmClass, and referenced in the alarm source:

- AlarmFunction [Simple(1), Basic(2), Extended Alarm(3)]
- AlarmClass [UrgentAlarm (1), HighPrioAlarm (2), NormalAlarm (3), LowPrioAlarm (4), UserDefinedAlarm (5) and OffLineTrend (6)]

Formula

The notification class number is calculated as follows:

NotificationClass# := 10 * AlarmClass + AlarmFunction

This gives the following notification classes:

| AlarmClass | AlarmFunction | Priority (default values) To-Offnormal To-Fault To-Normal | Uses | NotificationClass# (derived) |
|-------------|---------------|---|---|---------------------------------|
| | | | Highly critical alarms, system messages, device info object | |
| UrgentAlarm | Simple | 1, 1, 5 | | 11 |
| UrgentAlarm | Basic | 1, 1, 5 | | 12 |
| UrgentAlarm | Extended | 1, 1, 5 | | 13 |
| | | | Critical alarms | |

| AlarmClass | AlarmFunction | Priority (default values) To-Offnormal To-Fault To-Normal | Uses | NotificationClass# (derived) |
|------------------|---------------|---|--|---------------------------------|
| HighPrioAlarm | Simple | 2, 2, 6 | | 21 |
| HighPrioAlarm | Basic | 2, 2, 6 | | 22 |
| HighPrioAlarm | Extended | 2, 2, 6 | | 23 |
| | | | Normal alarms | |
| NormalAlarm | Simple | 3, 3, 7 | | 31 |
| NormalAlarm | Basic | 3, 3, 7 | | 32 |
| NormalAlarm | Extended | 3, 3, 7 | | 33 |
| | | | Non-critical alarms | |
| LowPrioAlarm | Simple | 4, 4, 8 | | 41 |
| LowPrioAlarm | Basic | 4, 4, 8 | | 42 |
| LowPrioAlarm | Extended | 4, 4, 8 | | 43 |
| | | | As project-specific alarms for special applications | |
| UserDefinedAlarm | Simple | 5, 5, 9 | | 51 |
| UserDefinedAlarm | Basic | 5, 5, 9 | | 52 |
| UserDefinedAlarm | Extended | 5, 5, 9 | | 53 |
| | | | Offline trends The To-Normal priority must be such that it is less than or equal to the Alarm Priority Limit of the device object (for Remote Mgmt) | |
| OffLineTrend | Simple | 2, 2, 2 | | 61 |
| OffLineTrend | Basic | 2, 2, 2 | | 62 |
| OffLineTrend | Extended | 2, 2, 2 | | 63 |

Project-specific notification classes can be defined in addition to predefined ones. Alarm classes 7...16 are intended for this purpose. The associated calculation of a notification class number is identical to calculation of predefined notification class numbers.

Customized alarms can be engineered in Desigo PX. In this case, the value for a notification class number can be defined without restrictions.

Priority [Prio]

This defines the alarm priority on the basis of which alarm and system events are to be transmitted to the receivers. Every transition can be described individually with this BACnet property, data type ARRAY of INTEGERS [TO_OFFNORMAL; TO_FAULT; TO_NORMAL]. Priority levels can range in value from 0 to 255. The lower the value, the higher the priority. In Desigo only priorities 1 to 9 are used.

Alarmfunction [AlmFnc]

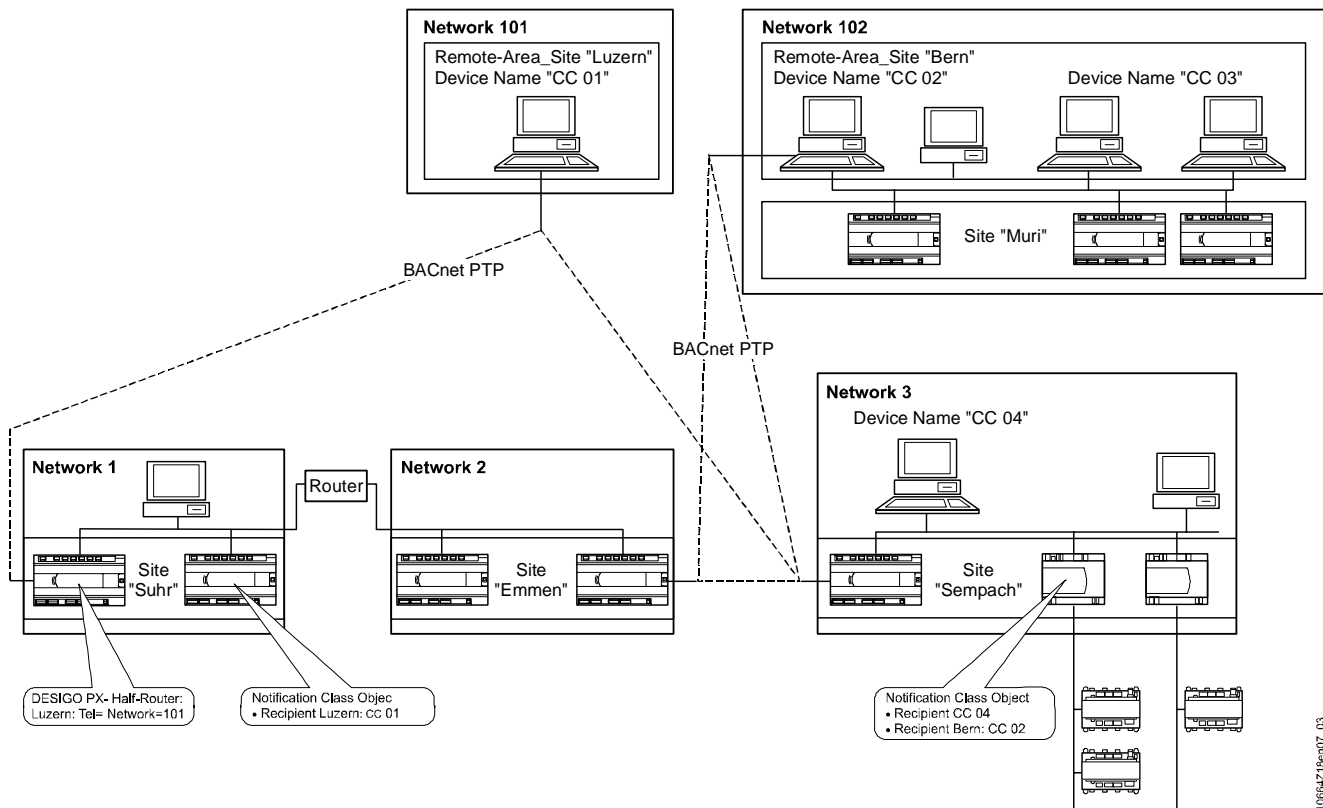
Alarm function types: Simple, Basic or Extended. [AlmFnc] is only supported by Desigo PX.

Destination list [RecpList]

The configured (permanent) alarm recipients, the week days, and the time window in which the alarm recipient is operated, are entered here. [RecpList] is equivalent to the standard BACnet property Recipient_List.

Destination list [DestLi]

This is where the configured (permanent) alarm receivers are entered, together with the days of the week and the time-window in which the alarm receiver is operated. [DestLi] is only supported by Desigo PX.



10664Z18en07_03

Operator units:

- Permanently connected operator units (and hence, alarm receivers) are addressed by their Device Name.
- Operator units (and hence alarm receivers) with a point-to-point connection (PTP connection) are addressed with a Remote Area Site identifier and their Device Name. For example:

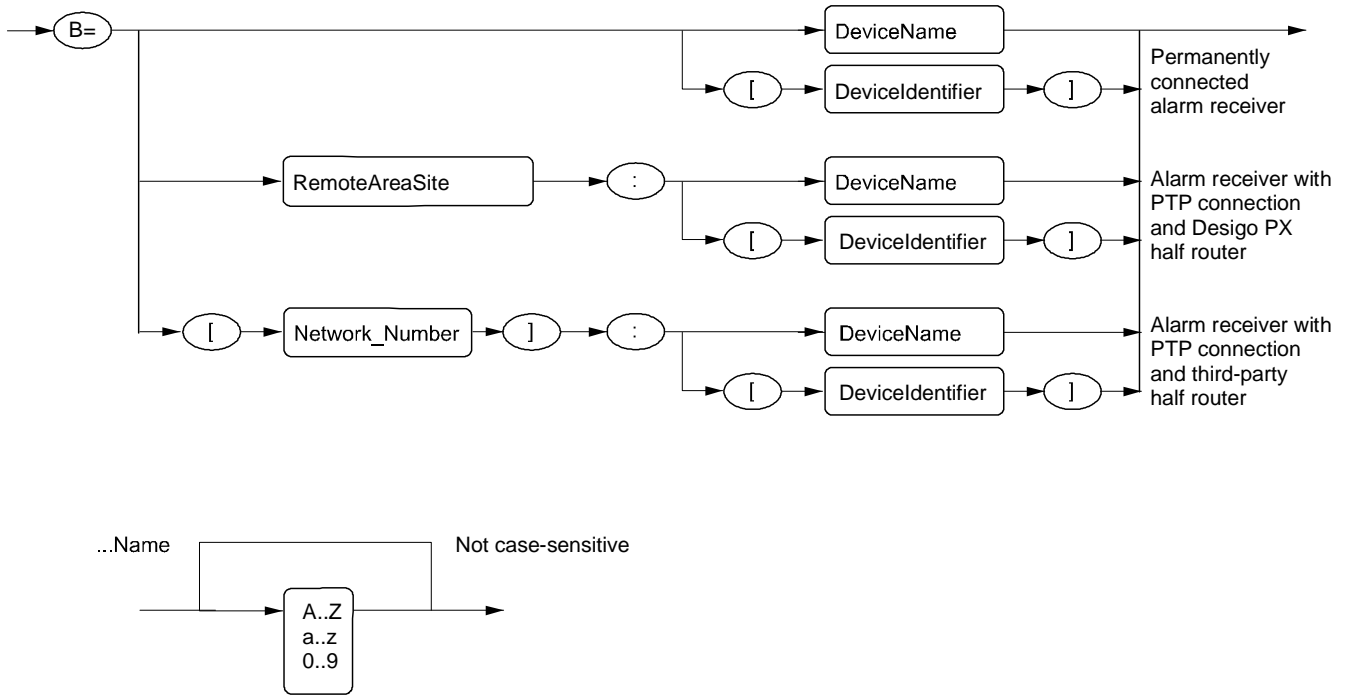
B=fff for permanent connection

B=kkk:aa for point-to-point connection (PTP connection)

- Adjustments are required during the addressing process so that there is no conflict between the names of operator units and the plant or room management designations.

Permanent and point-to-point connections:

- For alarm receivers, the address syntax (see further below) indicates the type of connection: permanent or PTP connections.
- Desigo PX automation stations with half-routers must know the Remote Area Site designators of their remote alarm receivers to enable an PX automation station to resolve the remote alarm receiver designator.



Alarm receiver syntax:

| Element | Description |
|--------------------|---|
| DeviceName | Device name. In plain text so that the user can understand it. Example: CC 01 |
| DeviceIdentifier | Device Identifier. Alternative syntax for the alarm receiver of a third-party manufacturer. If the alarm receiver has a special address range or if DeviceName does not work. Example: [13456] |
| RemoteAreaSiteName | Remote area site name. In plain text so that the user can understand it. Example: Chur |
| NetworkNumber | Network number. Required with a third-party half router. Example: [3] |

9.7 Alarm routing over the network

Alarm server and alarm clients

Alarm servers are entities capable of producing an alarm. Alarm clients are entities capable of receiving an alarm.

There are two types of alarm client: temporary alarm receivers and pre-configured alarm receivers. The following concept for temporary alarm recipients is only valid for Desigo PX.

Temporary alarm receivers

Temporary alarm receivers are not defined at the engineering stage. They can be connected to or removed from the network at any time during operation. If a temporary alarm receiver is connected to the network, it will perform the following activities for every alarm server:

- The alarm recipient enters its address in the BACnet property recipient list [RecpList] of the BACnet device object of the automation station, using the BACnet service AddListElement.
- Read information about all currently existing alarms, and all currently outstanding acknowledgements, from the automation station (BACnet service GetEventInformation). This ensures that the alarm receiver – irrespective of when it was connected – displays the current alarm status of the system.

After making these entries, the temporary alarm receiver, while connected, will receive all alarm messages from the automation station in accordance with the routing mechanisms described below.

If an automation station cannot transfer an alarm message to a temporary alarm receiver (e.g., because it is no longer connected to the network), the address of the receiver concerned will be removed from the [RecpList]. All alarm messages destined for that receiver will then be deleted.

Preconfigured alarm receivers

The preconfigured alarm receivers are entered in the notification class object:

- In the [DestList] for Desigo PX
- In the [RecpList] for Desigo room automation

Time response in the network

The routing of all alarm and acknowledgement messages between the alarm server and the alarm clients takes place over the BACnet network using special BACnet services. These are:

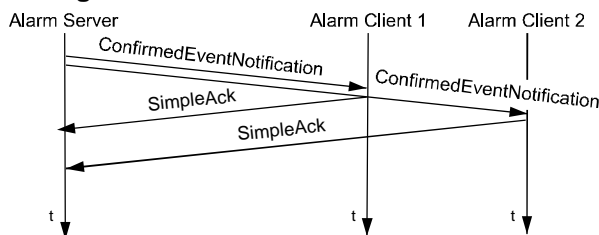
- Confirmed Event Notification for all changes in the alarm state of an alarm-generating object (TO_OFFNORMAL, TO_NORMAL, TO_FAULT), and for messages via local acknowledgements. Direction: From alarm server to alarm client.
- AcknowledgeAlarm for the routing of acknowledgements (including reset) performed by the user on an alarm client. Direction: From alarm client to alarm server.

The two services are referred to as Confirmed Services, that is, the receiving device always confirms the receipt of a service by immediately returning a SimpleAck message. This tells the transmitting device that its message has been received by the receiving device. If no SimpleAck message is received, the transmitting device tries to send the message again (up to three times).

An alarm is always issued by one (and only one) alarm server. Generally, however, there will be several alarm clients on the network. To maintain consistency, all alarm clients must always display the same alarm state. For this reason, all alarm-related functions must always be routed to all the alarm clients. The procedure is the same for both temporary and pre-configured alarm receivers.

The following time-diagrams describe the communications via the network for the various alarm-related events.

Change of alarm state



This procedure is carried out for every change in alarm state on an alarm server: TO_OFFNORMAL, TO_FAULT and TO_NORMAL. The data record Confirmed Event Notification contains the following information:

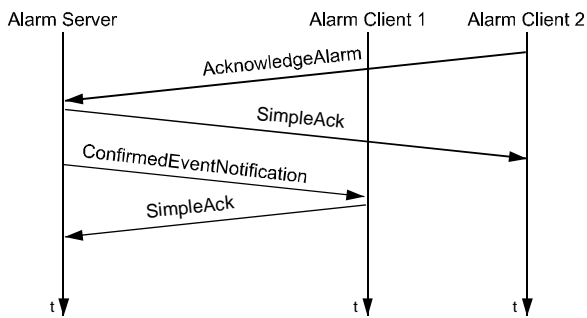
- BACnet address of the alarm server
- Object ID of the alarm-generating object
- Time stamp
- Alarm priority
- Initial and final state of the transmitted state transition (this is used to determine whether the state transition is TO_OFFNORMAL, TO_FAULT or TO_NORMAL)
- Acknowledgement required [AckReq]: Does the notified state transition require acknowledgement or not?
- Alarm text
- Other technical details

Based on this information, the alarm client can present the alarm in a comprehensible way; it may also read additional information automatically from the alarm server, and if required, return any acknowledgement to the correct address.

If a temporary alarm receiver does not confirm receipt with a SimpleAck message (via the Confirmed Event Notification input), the alarm server will try three times more to transmit the alarm to the relevant alarm receiver. The message for this alarm client will then be lost and its reference will be deleted from the [RecpList] of the BACnet device object.

Alarm acknowledgement over the network

This process is performed for all acknowledgements made on an alarm client.

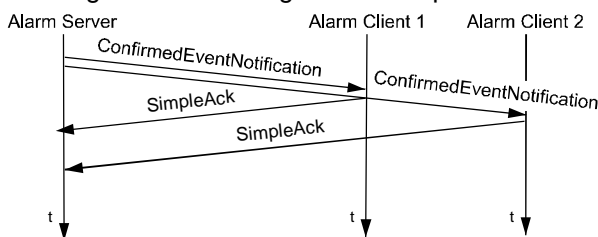


Acknowledge and reset

The alarm can be acknowledged by any alarm client. The AcknowledgeAlarm data record contains information as to which alarm is being acknowledged and other details related to this alarm and the acknowledging alarm client. The alarm acknowledgement is confirmed with a SimpleAck message by the alarm server which generated the alarm. All other alarm clients in the network will be sent a Confirmed Event Notification to notify them of the alarm acknowledgement. They, in turn, will send a SimpleAck message to acknowledge the receipt of this notification, the objective being to ensure that all clients have up-to-date and consistent information.

Local alarm acknowledgement

Alarms can also be acknowledged and reset locally on the alarm server. The alarm is acknowledged internally in the alarm server that generated the alarm. Confirmed Event Notifications are now transmitted to all alarm clients, to notify them that the alarm has been acknowledged. The alarm clients, in turn, send a SimpleAck message to acknowledge their receipt of the Confirmed Event Notification.



Disabling the routing of alarms

Each alarm-generating object has an [EnEvt] parameter (data type: Boolean). Alarm messages (and system events) are only transmitted over the network if [EnEvt=TRUE]. This does not affect the alarm monitoring of the object, that is, the alarm state machine is always kept up to date.

9.8 Alarm queuing

Until they have been forwarded to all the pre-configured alarm recipients, all alarm messages are normally stored in the automation station.

Each automation station has its own alarm queue for this purpose. Each incoming alarm and each system event is entered in the queue. An entry remains in the queue until the alarm or system event has been sent with a confirmed event notification to all recipients listed in the notification class object, and until the relevant acknowledgements have been received.

If the queue is full or overflowing, the oldest entries are deleted automatically, and a system event message is generated. Entries are deleted irrespective of alarm priority.

Alarm queuing has no effect on the alarm state of the alarm source.

Alarms destined for a temporary alarm recipient are not saved in the automation station. If a temporary alarm recipient can no longer be reached, the address of the recipient concerned is removed from the [RecpList] of the device object.

BACnet device object properties

The queuing of alarms is controlled by the following BACnet properties in the device object of the Desigo PX automation stations. These properties are not mapped to a function block, and can therefore only be viewed and modified online.

Buffer size [BufSize]

This BACnet property defines the maximum number of entries which can be saved in the queue.

A new value will only be accepted if it is greater than the record count [RecCnt].

Buffer size [BufSize] of the alarm queue.

- Default = 100 (PXC) or 150 (PXR)
- Range = 10...500 depending on the available memory space

Record count [RecCnt]

This BACnet property represents the number of entries currently stored in the queue.

The alarm queue can be deleted by writing the value 0 to this property. A write of a value not equal to 0 results in an error message.

If the queue is deleted, this information is entered as a system event in the queue and transmitted to the receivers. This causes the value to change to 1 as soon as [RecCnt] is set to zero.

Notification threshold [NotifThd]

This BACnet property defines the dial-out threshold, and the number of alarms to be deleted in the event of a queue overflow.

If [RecCnt] is greater than or equal to [NotifThd], a connection is established with any alarm recipients connected by cable (modem) which are destined to receive alarms and events from the queue. The connection is established provided that the remote alarm recipient concerned is listed in the notification class object.

The message threshold also defines how many alarms are to be deleted in the event of a queue overflow. As many alarm entries are deleted as necessary until [RecCnt] is equal to [NotifThd]. This function does not distinguish between local and remote alarm recipients in the notification class object.

To avoid deleting too many alarms, it is recommended that [NotifThd] be set to approximately 80% of the [BufSize].

Alarm queue message threshold [NotifThd]:

- Default = 80 (PXC) or 130 (PXR)
- Range = 5...495

Alarm priority limit [PrImAlm]

This BACnet property defines another independent threshold for dial-out.

An alarm priority which is less than or equal to [PrImAlm] results in a connection with any alarm recipients connected by cable (modem) which are destined to receive alarms and events from the queue. The connection is established provided that the remote alarm recipient concerned is entered in the list of Notification Class objects.

Alarm priority limit [PrImAlm]:

- Default = 2 (HighPrioAlarm or UrgentAlarm)
- Range 0...255

If the notification class object contains only local alarm recipients, then the optimum results are achieved by use of the default values for control of alarm queuing. The default values should therefore not be changed.

If the notification class object contains remote alarm recipients, then it may be appropriate to modify the default values for control of alarm queuing.

The values for [NotifThd] and [PrImAlm] determine when a remote cable connection (modem) is to be established, in order to inform the user of the occurrence of alarms.

If low-priority alarms are to be forwarded immediately, the [PrImAlm] value must be increased (the higher the number, the lower the alarm priority). The value for [NotifThd] must not be modified.

The value for [NotifThd] can be reduced, however, in cases where a connection is to be established when there is a smaller number of alarms in the alarm queue. It is important to ensure, however, that the difference between [BufSize] and [NotifThd] does not become too great, as this is the value that controls the deletion of alarms in the event of a queue overflow.

Note that modifying these values also affects connection costs.

It takes time to establish a connection by cable (modem). If it is likely that further alarms will occur during this time, thereby causing the queue to overflow, the difference between [BufSize] and [NotifThd] should be increased.

The following settings are recommended:

- Buffer size [BufSize] = 120
- Notification threshold [NotifThd] = 80

In cases of doubt, the default values should be left unchanged.

9.9 Common alarms

The BACnet object alarm states InAlarm, Unacked and Unreset are grouped in the following blocks:

- The CommonAlarm block for Desigo PX
- The CommonEvent block for Desigo room automation

The difference between CommonAlarm and CommonEvent is, that the CommonAlarm block supports Intrinsic Reporting. The alarm detection and notification of the CommonEvent block is handled by a special Event Enrollment object called CommonEventEnrollment. The CommonEventEnrollment block also handles the common alarm reset / ack and common manual intervention functions.

All alarms generated by alarm-generating BACnet objects on the same chart level or subordinate charts are automatically grouped into a common alarm. There is therefore no need for the user to create a common alarm by establishing links or interconnections. The engineering process simply involves placing the block at the required chart level. No other configuration steps are necessary.

Common alarm reset / ack

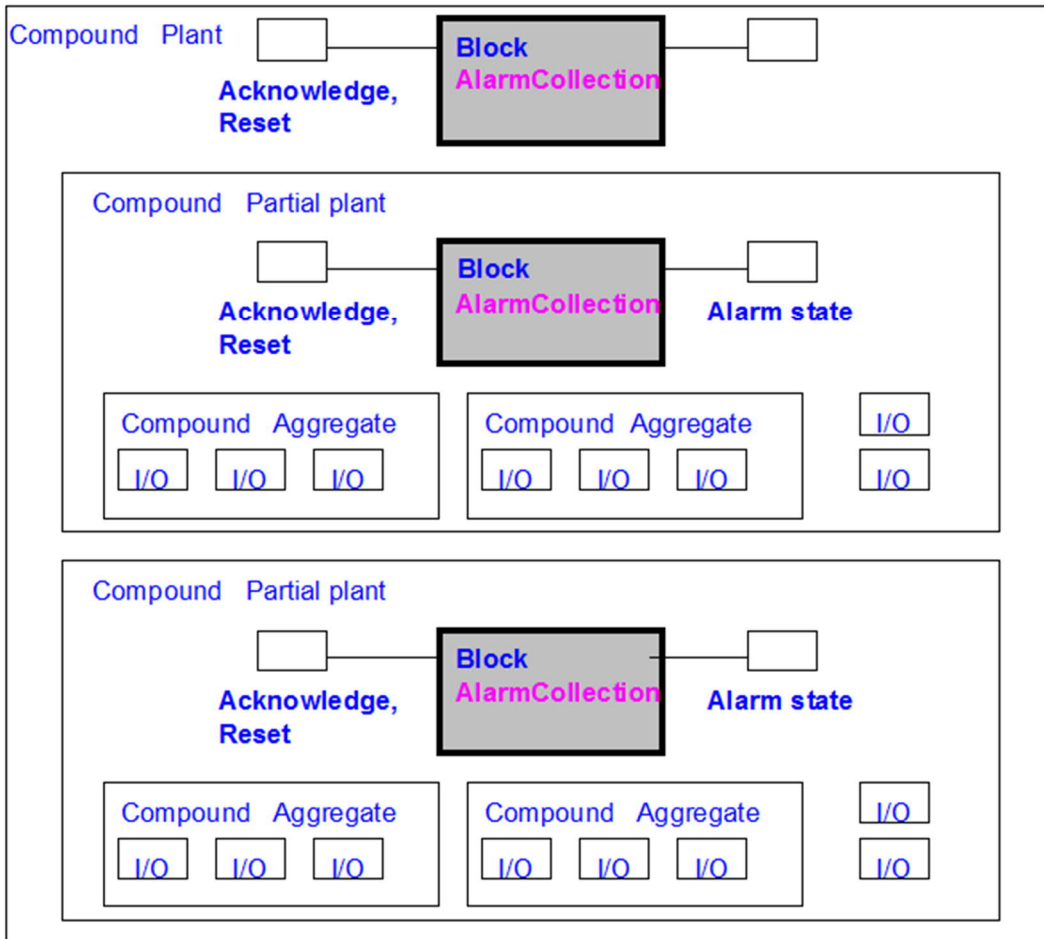
Similarly, all the alarms covered by this block can be the subject of a common alarm reset and acknowledge. Acknowledging the common alarm object is equivalent to acknowledging all objects on the same and lower levels in the hierarchy.

Resetting the common alarm object is the equivalent to resetting all objects on the same and lower levels in the hierarchy.

Common manual intervention

The same common alarm object also uses the status flag Overridden to indicate the manual operation of one or more of the BACnet objects (with [StatFlag] override facilities) on the same or a lower chart level. Manual intervention are determined on the properties: Out of service [OoServ], overridden, commanding to Prio 7 (manual switch) and Prio 8 (operator).

This diagram shows the practical application of the common alarm object within the technical hierarchy. The common alarm object in the partial-plant compound encompasses all the alarms of this partial plant. The higher-level common alarm encompasses the alarms of both partial plants.



9.10 Alarm suppression

Alarm suppression refers to suppression of alarm and event notifications in the Desigo system. Thus, sending BACnet event notifications is suppressed. Alarm suppression does NOT prevent detection of alarm states.

Alarm suppression types

The following types of alarm suppression exist in Desigo:

- Alarm suppression by automation station using function block AS_STA allows for implementing alarm suppression at the automation station level.
- Hierarchical alarm suppression is made possible via the common alarm object based on the structure of the technical view.
- Specific alarm suppression: All alarmable objects offer alarm suppression by object.
- Exceptions for alarm suppression: Each alarmable object has a pin SupEcpt. This pin allows for defining exceptions to hierarchical alarm suppression.

Validity for alarm suppression

All types of alarm suppression apply to Desigo PX. Desigo room automation devices can generate and suppress alarms.

Alarm suppression by automation station

AS_STA (Device Access) is a Desigo PX function block that allows to suppress all alarms of an automation station. The function block allows for suppressing BACnet event notifications by means of an application. Thus, sending of alarms and events during, e.g., maintenance, can be suppressed, e.g., via a key switch.

Alarm suppression is controlled via pin SupEvt. The following values are defined for SupEvt:

- true: The automation station sends NO BACnet event notifications.
- false: The automation station sends BACnet event notifications.

For more information on function block AS_STA, see:

Desigo Firmware blocks, automation level, Overview (CM110749)

Desigo Vxx Firmware blocks (CM110729)

Desigo room automation supports the suppression of all alarms of an automation station. To do this, it uses the device infrastructure objects CommonEvent and CommonEventEnrollment.

Hierarchical alarm suppression

Hierarchical alarm suppression allows for suppressing alarms of a plant, partial plant, aggregate, component, or subcomponent. Hierarchical suppression is based on suppressing alarms of any part of a partial tree in the technical structure.

For Desigo PX hierarchical alarm suppression is carried out via the Alarm Collection (CMN_ALM) object. CMN_ALM comprises all alarmable BACnet objects as a group on the same or lower hierarchies in the technical view. Thus, CMN_ALM allows for controlling alarm suppression for all alarmable BACnet objects of a group.

Alarm suppression is controlled via pin SupEvt. The following values are defined for SupEvt:

- true: Alarmable BACnet objects of a group do NOT send BACnet event notifications.
- false: Alarmable BACnet objects of a group send BACnet event notifications.

Desigo room automation supports the hierarchical suppression of alarms. It uses the CommonEvent and CommonEventEnrollment objects.

The CommonEvent object aggregates the alarm state of all BACnet objects on the same or the lower hierarchy levels of the technical view.

The CommonEventEnrollment object monitors the CommonEvent object. The hierarchical alarm suppression can be turned on or off in the CommonEventEnrollment object.

Specific alarm suppression

All alarmable BACnet objects allow for specific suppression. Each alarmable BACnet objects offers pin EnEvt for this type of alarm suppression.

The following values make sense for EnEvt:

- (False, False, False): The object sends NO BACnet event notification.
- (True, True, True): The object sends BACnet event notification.

Value combinations with True and False for EnEvt should be avoided.

Alarm suppression exceptions

A possible application is to activate alarm suppression during plant maintenance. Vital alarms must be excepted from alarm suppression.

For Desigo PX exceptions can be defined for hierarchical alarm suppression with CMN_ALM.

Function block CMN_ALM has pin EnSupEcp. This exception specifies if exceptions are possible within the alarmable BACnet objects of the group. The following values are defined for EnSupEcp:

- true: Exceptions for alarm suppression within the group of alarmable BACnet objects are considered.
- false: Exceptions for alarm suppression within the group of alarmable BACnet objects are NOT considered.

Each alarmable BACnet object can be exempted from hierarchical alarm suppression with CMN_ALM. To do this, each alarmable BACnet object has pin SupEcp. The following values are defined for SupEcp:

- true: The object is considered as an exception for alarm suppression.
- false: The object is NOT considered as an exception for alarm suppression.

Combination of multiple alarm suppressions

The above options to suppress alarms can overlap. For an object impacted already by multiple types of suppression, the following rule applies: One type of alarm suppression cannot be overridden by another type of alarm suppression.

| AS_STA. SupEvt | CMN_ALM. SupEvt | CMN_ALM. EnSupEcp | FB.SupEcp | FB.EnEvt | Resulting alarm suppression for function block |
|-------------------|--------------------|----------------------|-----------|-----------|---|
| True | • | • | • | • | suppressed |
| • | • | • | • | (F, F, F) | suppressed |
| False | False | • | • | (T, T, T) | not suppressed |
| False | True | False | • | (T, T, T) | suppressed |
| False | True | True | True | (T, T, T) | not suppressed |
| False | True | True | False | (T, T, T) | suppressed |

9.11 Alarm message texts

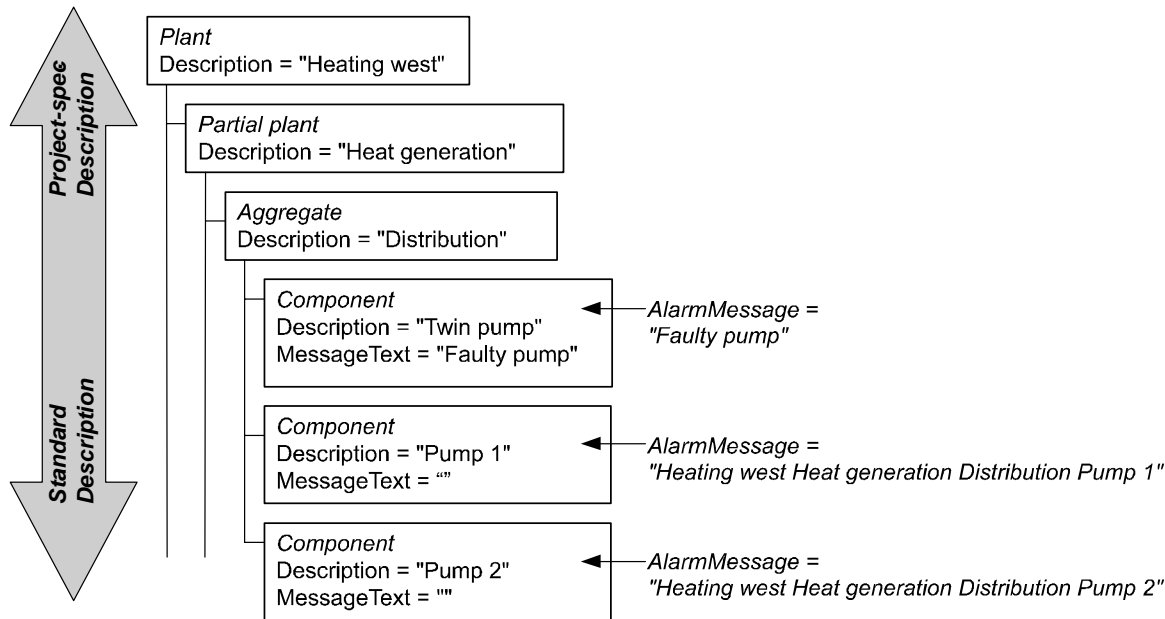
Desigo contains all the alarm texts necessary to help the user maintain an overview and an understanding of the alarms. These alarm texts can be freely defined in the engineering tool for each alarm source individually. If this is not done, text will be generated automatically on the basis of the Technical Designation of the individual function blocks. In a third category are the predefined alarm texts used in conjunction with device faults.

Configured alarm message texts

The Desigo system supports alarm message texts.

For Desigo PX the message texts for TO_OFFNORMAL, TO_FAULT and TO_NORMAL alarms are entered as an Array [3] in the BACnet property [Message_Text]. For Desigo room automation the BACnet Property Event_Message_Texts_Config is used.

For Desigo PX if no alarm message text has been entered for an alarm source, or if the alarm message text for a given state (e.g., TO_FAULT) has been left blank, an alarm message text will be generated from the existing descriptions.



Longer messages are divided into segments with forward slashes // so that the client can display the message over several lines. Each segment may contain a maximum of 70 characters, with a maximum of three segments separated by // for any one message.

Non-configured message texts

System alarms and events of the BACnet Device Info Object use text messages which cannot be configured, e.g., Battery low.

Predefined, language-dependent text

System alarms and events of the BACnet Device Info Object use non-configured text messages whose contents are language-dependent. These language-dependent texts are organized into text groups with a predefined server system text scope, and can be translated. The translated text groups are loaded into the automation station via BACnet description information.

10 Calendars and schedulers

Standard BACnet objects

The standard BACnet objects Schedule and Calendar are used for time scheduling functions in the Desigo system. These objects can be used to configure and operate time scheduling functions at different operating levels within the system and via BACnet-compatible operator units from other manufacturers.

The local PXM10 operator unit can also be used to operate the standard BACnet objects for the connected automation stations and PXC.

Function blocks

The time scheduling functions are implemented as function blocks in CFC charts of automation stations. Each automation station and each switching operation requires one schedule block. The pins of the function blocks are mapped to standard BACnet properties.

There are four versions of the schedule block, with an analog, binary or multistate output or with a variable data type (Boolean, Unsigned, Real or Enumerated). A schedule block can only contain schedule values of the same data type.

[WeekSchd] [EcptSchd]

The scheduler program consists of a weekly schedule [WeekSchd] and an exception schedule [EcptSchd]. The weekly schedule contains a 24-hour profile for each day. The exception schedule contains up to 20 profiles, which can be activated for a date or date range. The date or date range can be defined both in the schedule itself and in the calendar object.

[Prio]

A priority must be assigned to each of the profiles in the schedule. Based on the priority level assigned, the scheduler program determines from the priority [Prio] which profile is to be processed. The weekly schedule has the lowest priority.

[EfPrd]

The effective period [EfPrd] property defines the time period for which the schedule is active.

[PrVal] [NxVal] [NxTi]

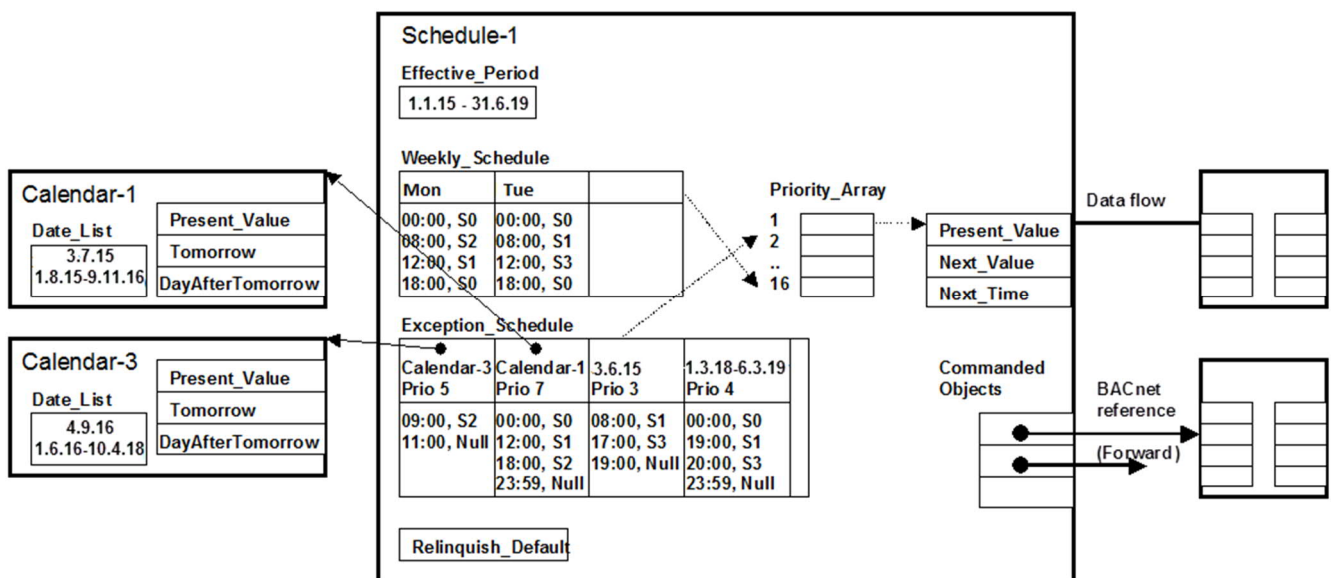
The present value [PrVal], next value [NxVal] and next time [NxTi] are available at the output of the time schedule. [NxVal] and [NxTi] are used for optimization purposes.

Commanded objects

The time schedule also incorporates a list of references to BACnet objects and (optionally) a property which is to be controlled by the scheduler program via BACnet.

[DefVal]

The BACnet property Relinquish_Default is the default value [DefVal] for the present value output [PrVal].



10.1 Schedule

Weekly schedule [WeekSchd]

The weekly schedule [WeekSchd] consists of seven 24-hour profiles, one for each day of the week. By default, the priority level assigned to the weekly schedule is 16 (the lowest priority). The weekly schedule is active unless there is an exception schedule.

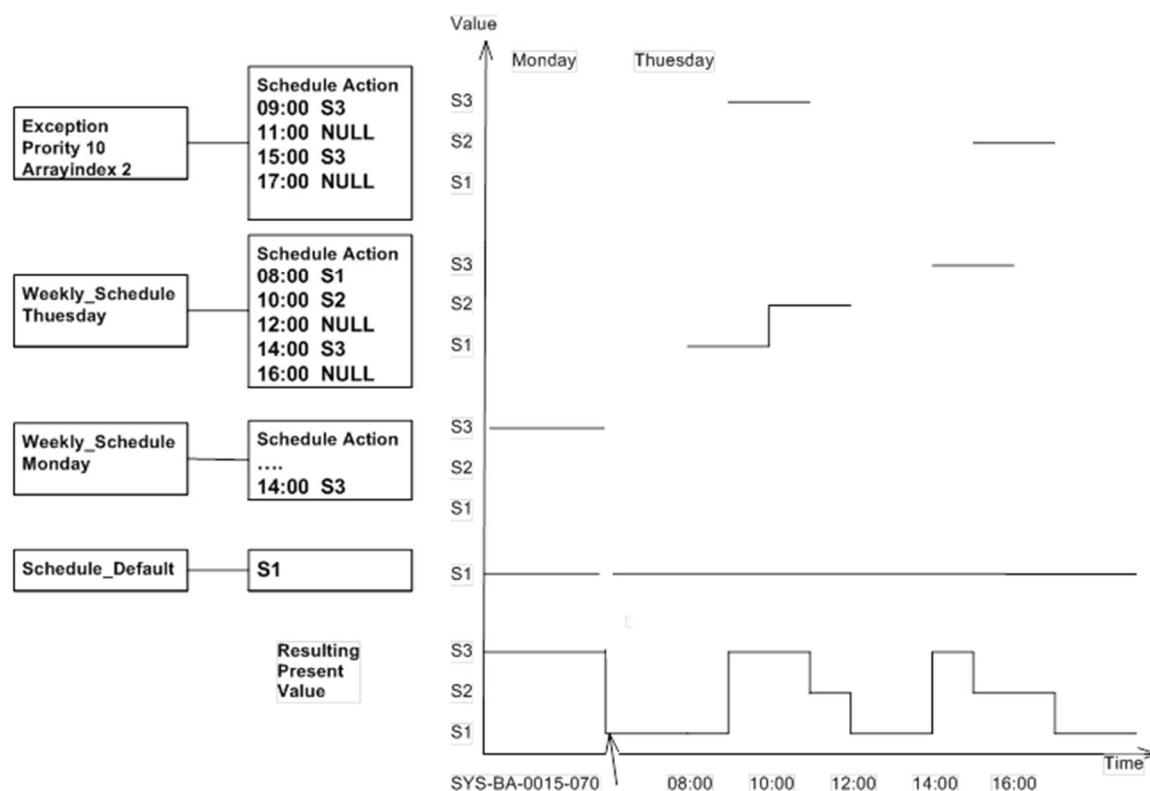
For system limits, see chapter *System Configuration*.

24-hour profiles

A 24-hour profile is a list of time-and-value pairs. The present value remains at the [PrVal] output until the processing of the next time-and-value pair causes a new value to be written to the output.

If there is no schedule entry with a switch time of 00:00 in the daily profile, the default value determines the resulting Present_Value (=Rule schedule default value).

If the daily profile encompasses an empty list of schedule entries, the default value [DefVal] determines the resulting Present_Value (=Rule schedule default value).



The evaluation of the exception schedule, weekly schedule, and [DefVal] is as follows:

Start of the day:

- Exception schedule with switch value at 00:00:
The exception schedule determines the resulting Present_Value if an active switch value exists at 00:00. The day begins with this exception value (=Rule switch value exception schedule).
- Empty daily profile:
If the daily profile encompasses an empty list, the default value [DefVal] determines the resulting Present_Value (=Rule schedule default value).
- Daily profile with switch value at 0:00.
If a schedule entry with switch time 00:00 and active switch value is available in the daily profile, the switch value determines the resulting Present_Value. The day begins with the daily profile value (=Rule Switch value daily profile).

Course of the day:

- Switch value exception schedule:
If an active switch value exists for a specific time, the exception schedule determines the resulting Present_Value.
- Daily profile switch value:

If an active switch value from a daily profile exists for a specific time, the daily profile determines the resulting Present_Value.

- Default value switch value:

If no active switch value from the exception schedule and the daily profile exists at a specific time of day, the default value determines the resulting Present_Value.

Exception schedule [EcptSchd]

Exception profiles

The exception schedule [EcptSchd] overwrites some or all of the daily switching operations in a weekly schedule [WeekSchd]. It consists of one or more profiles (max. 20).

Each profile has a:

- Date
- Specified time
- Priority
- Value for the output signal

The exception schedule may be a time range, including multiple days.

Activating exception profiles

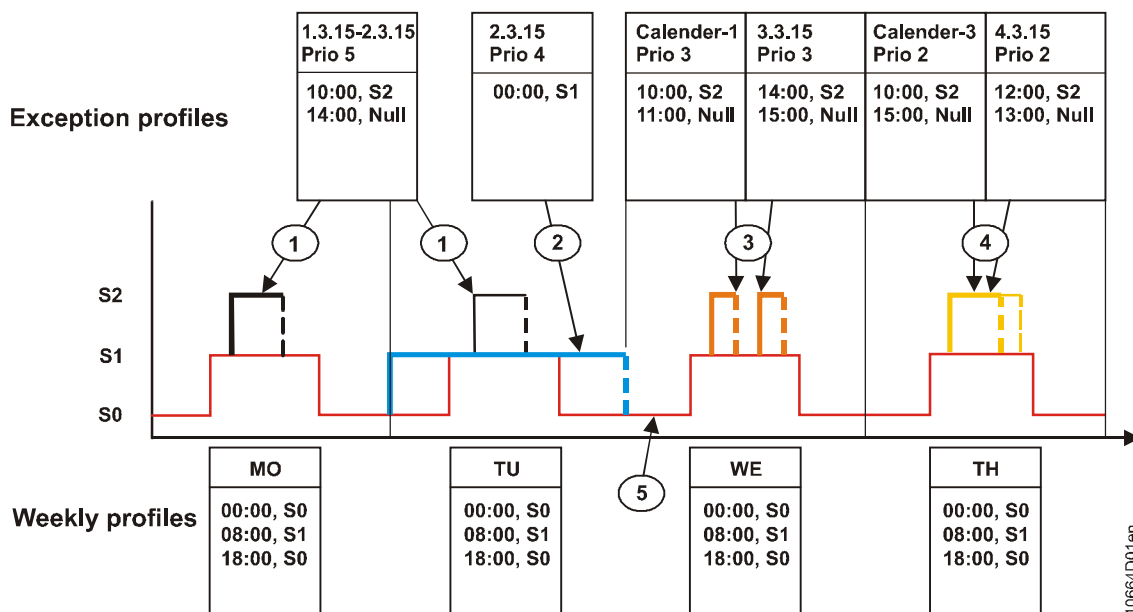
Depending on the customer's requirement, the date on which an exception profile is to be activated can be defined either in the time schedule itself, or in the standard BACnet object Calendar. In the latter case, the calendar object is linked to the time schedule via BACnet references.

An exception begins with the first time entry and ends with the last. Each profile may contain up to 20 switch times.

Setting priorities

The switch value of all current profiles is continuously monitored for present priorities. The priorities determine which switch value is transferred to the [PrVal] output. The system evaluates every minute if a day or an exception profile should be active. Each profile of the exception schedule is assigned a priority level from 1 (highest) to 16 (lowest). If several exceptions are valid at the same time, the profile with the highest priority is processed.

If multiple switch values from different exceptions with the same priority exist at a specific time, the active switch value of the exception with the lowest array index (from the exception schedule) determines the exception switch value. The procedure is the same as the procedure for various priorities. The array index is used as a sub-priority. Exceptions with different priority levels however, are independent of each other. That's why it is preferable to assign different priorities to exceptions defined in the schedule.



Key

- ① An exception profile applies to more than one day. On the second day, the exception profile is inactive, because another profile with a higher priority is active for the whole day.
- ② An exception program without the entry NULL. This exception profile is active for the whole day and ends automatically in the automation station at 24:00 hours by the NULL entry.
- ③ Several exceptions with the same priority on the same day, but without overlapping times. The exception profiles do not interfere with each other as an exception begins with the first time entry and ends with NULL.
- ④ Several exception profiles with the same priority on the same day with overlapping times. These exception profiles affect each other, as several exceptions with the same priority level are active simultaneously. In such cases, the rule is that if the switch commands are the same, the first time-entry applies (in this example 13:00 to NULL). With non-identical switch commands, the latest time-entry applies.
- ⑤ Operation in accordance with the weekly schedule.

Output signals

[PrVal] [NxVal] [NxTi]

The scheduler sends the following output signals:

- [PrVal]
- [NxVal]
- [NxTi]

The [NxVal] und [NxTi] output signals support the optimum start/stop control of the plant. When determining [NxVal] and [NxTi] in the time schedule, the current day and the next two days are taken into account. This results in a time window of 48 to 72 hours, depending on the current time and the next switch entry. If there is no change in [PrVal] within the time window, then [NxVal] is the same as [PrVal] and [NxTi] is equivalent to the current date plus 3 days (00:00h).

[DefVal]

This default value [DefVal] appears at the [PrVal] output when there is no active entry in the time schedule, or when the entries are all NIL, or when the time period is outside the active period.

[EnDef]

The [EnDef] variable enables or disables the [DefVal] variable.

The function block variables [DefVal] and [EnDef] are mapped to the Schedule_Default property. The property Schedule_Default can have the value [DefVal] or NIL.

| Variable DefVal | Variable EnDef | Property Schedule_Default |
|-----------------|----------------|---------------------------|
| Value | True | Value |
| Don't care | False | NIL (= Release) |

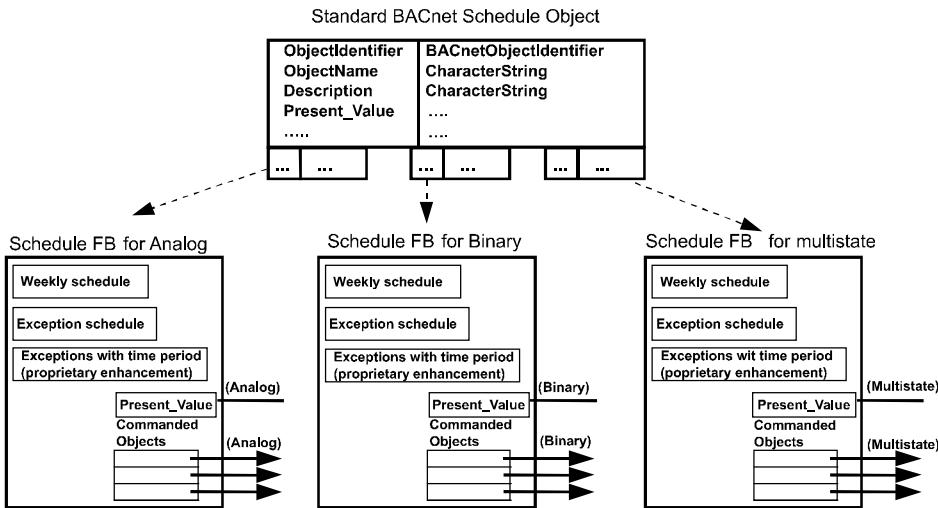
The NIL value in the Schedule_Default property is the release value for the active priority of the object controlled by the scheduler. Do not confuse it with the NIL value in the exception schedule used to prioritize the time entries.

Function blocks for various data types

There are four versions of the schedule block, with an analog, binary or multistate output or with a variable data type (boolean, unsigned, real or enumerated).

| Function block | Output | Example |
|--------------------------------------|--|--------------------------------|
| BSchd (PX) | Binary | True/False |
| ASchd (PX) | Analog | 20°C |
| MSchd (PX) | Multistate | Off, Stage 1, Stage 2, Stage 3 |
| Schd (PX and Desigo room automation) | Boolean / Unsigned / Real / Enumerated | |

The switching value is output to [PrVal] and to the objects to be switched (commanded objects list). A schedule block can only contain switching values of the same data type (binary or analog or multistate or boolean or unsigned or real or enumerated). It is therefore not possible to switch two different data types in sequence.



In Desigo PX the CAL (calendar) and SCHED (schedule) function blocks can be created online.

Commanded objects

The schedule can influence other commandable objects, irrespective of whether or not they are in the same automation station.

The schedule is thus a grouping object and contains a list of group members, in the form of a list of name references [NamrList]. These group members are the commanded objects, that is, the objects to be switched. The list can contain up to five entries.

Referencing

The referencing of group members is resolved at runtime.

Information flow

The grouping and the information flow only go in one direction (forward referencing).

The information flows inside one automation station or across several automation stations. The scheduler object recognizes the flow of information and knows where to send information and what data type is required by the group members. The information transmitted covers only the present value [PrVal] or the values for the Optimum Start/Stop functions [PrVal], [NxVal] and [NxTi].

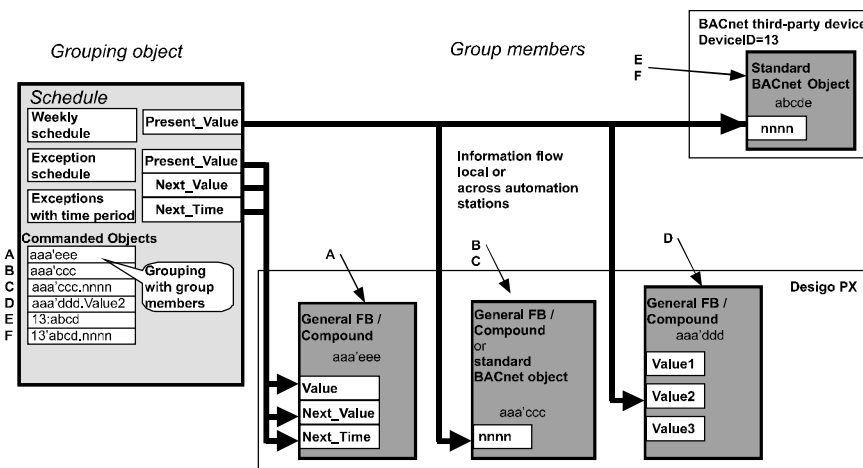
Heartbeat [Hrtbt]

In Desigo PX the function block variable Heartbeat [Hrtbt] determines the period measured in seconds at which the current value (Present_Value) is written.

Enable_Repeat_Command [EnRptCmd]

In Desigo PX the function block variable Enable_Repeat_Command [EnRptCmd] defines if the switching action is executed if the Present_Value does not change:

- EnRptCmd = TRUE: Switching action is executed if Present_Value does not change.
- EnRptCmd = FALSE: Switching action is NOT executed if Present_Value does not change.



Effective period [EfPrd]

You can define the period for which the schedule is to be active, e.g., you can configure separate schedules for summer and winter operation. If the current day is outside the active period, the [PrVal] output is equal to the default value [DefVal].

Time resolution

The smallest unit in the scheduler program is one minute and in the calendar one day. The schedule may be dependent on the calendar. In the PX automation stations, the calendar function block is automatically processed before the scheduler function block. The superposed cycle for processing the calendar and scheduler begins at the start of the new minute of the system time.

A PX automation station incorporates an automatic load shedding mechanism. The result is that a switch command at time x is executed within a time-period defined by $x + 1$ minute.

System time

Schedules and calendars are based on the same global time. This ensures that all automation stations on a site have the same time base.

Interdependency and order of processing

Interdependency of function blocks

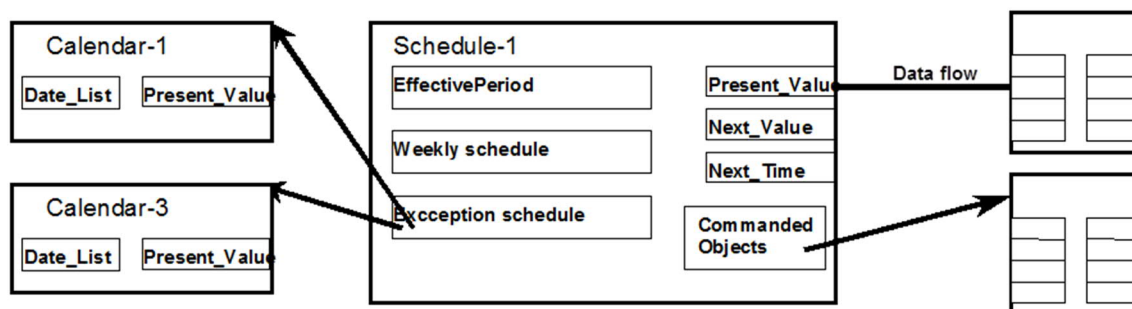
The calendar and schedule function blocks are standalone objects which are processed individually. The Schedule function blocks depend on the Calendar function blocks. The objects to be switched (commanded objects or data flow output) depend on the Schedule function blocks.

Processing order

At start-up, when delta loading and when adjusting the date and time, the order of processing is a key factor in ensuring that from the first processing cycle on, the correct output values of a schedule function block are determined and transmitted to the output. The temporary transmission of incorrect switch values can be avoided in this way. The order of processing of the individual function blocks is determined in the CFC Editor (manual/automatic).

The order of processing is:

1. Calendar function blocks
2. Schedule function blocks
3. Any other function blocks, which could be switched by a schedule function block



10.2 Calendar

Function block Calendar

The calendar object is a function block from the firmware library. It contains a list of dates [DateList] with, e.g., a date or a date range.

The date list [DateList] uses Boolean logic to control the calendar outputs. [PrVal] activates an exception profile if the calendar object is referenced by a schedule object. The outputs tomorrow [Tmw] and day after tomorrow [DayAfTmw] support the optimum start/stop control of the plant.

Standard BACnet object Calendar

The SCHED (schedule) and CAL (calendar) function blocks in the firmware library correspond to the SCHED and CAL standard BACnet objects. Standard BACnet object can be operated via the BACnet clients.

The calendar and schedule can be linked at the BACnet level by references. There is no data flow link between the calendar and schedule function blocks in the CFC chart.

10.3 Wildcards

A wildcard character (*) generates a repetition and is an abbreviated way of listing individual entries, e.g., writing 3.* is a short way of representing 3.1., 3.2., 3.3., 3.4., 3.5., etc.

All data structures of the scheduler or calendar objects support dates with wildcards. Date ranges and time specifications do not support wildcards. Invalid weekdays are ignored.

Date entries with wildcards

| Date | Meaning |
|------------------------|---|
| 23.April.2001 /Monday | 23.April.2001, Monday |
| 23.April.2001 /Tuesday | Never, since 23.April 2001 is a Monday |
| 23.April.2001 /* | 23.April.2001 |
| 23.April.* /Monday | Each April 23rd, each year if the weekday is a Monday |
| *.April.2001 /* | Every day in April 2001 |
| *.April.* /Tuesday | Each day in April of each year if the weekday is a Tuesday |
| 31.*.* /* | Each January 31, March 31, May 31, ... of each year or each February 28/ 29, April 30,... of each year |

If a date contains a wildcard in the month or year, the last day of the month is used for the day, if the value of the day is greater than the maximum number of days in the month.

Week and day with wildcards

The following table shows an example of entering a week and day (WeekNDay) using wildcards. During the evaluation, a wildcard is replaced by the corresponding value of the current date. If the WeekNDay generated in this way is equivalent to the current date, this is an exception day.

| Day of the week | Meaning |
|----------------------|--------------------------------------|
| January/2/Monday | Monday in the second week of January |
| */1/Tuesday | Every first Tuesday of a month |
| February/*/Wednesday | Every Wednesday in February |

10.4 Alarm messages

The scheduler object cannot directly generate alarms when, e.g., a commanded object cannot be found.

The alarm function (extended, basic or simple) which defines the alarm behavior of the object in the system, and the alarm class do not exist because of standard compliance.

The fault alarming of the schedule object must therefore be implemented as an additional binary value function block. That's why the function block is linked with the Dstb (Disturbed) pin of the scheduler and parameterized accordingly. The additional binary value function block is optional and only required, when a fault alarming of the scheduler is wanted, e.g., for scheduling external objects.

11 Trending

Trend data provide important information about the processes in a building automation and control system, e.g.:

- Monitoring of the control system for optimization purposes
- Logging the room temperature in association with the set temperature
- Logging of temperature and humidity trends for the pharmaceutical industry

Offline/Online trend

There are two types of trend data:

- **Offline trend:**
The recorded trend data are saved in the automation station and uploaded to Desigo CC periodically or as needed. The data can be analyzed in Desigo CC.
A connection is needed only during the data upload. Trend objects are needed in the automation station.
Offline trend is mostly used for long term data logging.
- **Online trend:**
Arbitrary data points can be saved as online trends.
A permanent connection is needed. No trend objects are needed in the automation station.
Online trend is mostly used for temporary data logging.

Trend Log Objects

The trend data is saved in the buffer of the Trend Log and Trend Log Multiple objects in the automation station. The Trend Log object can only record one value of a data point. The Trend Log Multiple object can record up to six different values of a data point.

The Trend Log object cannot be set up online, but must be set up in advance, offline in the application. A Technical Designation (TD) determines (BACnet reference) which object is to be logged. This presupposes that the referenced object is visible via BACnet (not No Element). The reference and the parameters can be defined and modified online or offline.

When the number of trend log entries reaches a definable threshold (Notification Threshold [NotifThd]), the Trend Log object generates an event. The Trend Log object sends out an alarm which is defined in a notification class specified for Trend Log.

The trend log data acquired can only be read, and if necessary, archived, with a BACnet Client configured for this purpose, e.g., Desigo CC. The status of a Trend Log object is not affected by reading out trend log data. It is not possible to reload sampled data into Xworks Plus (XWP).

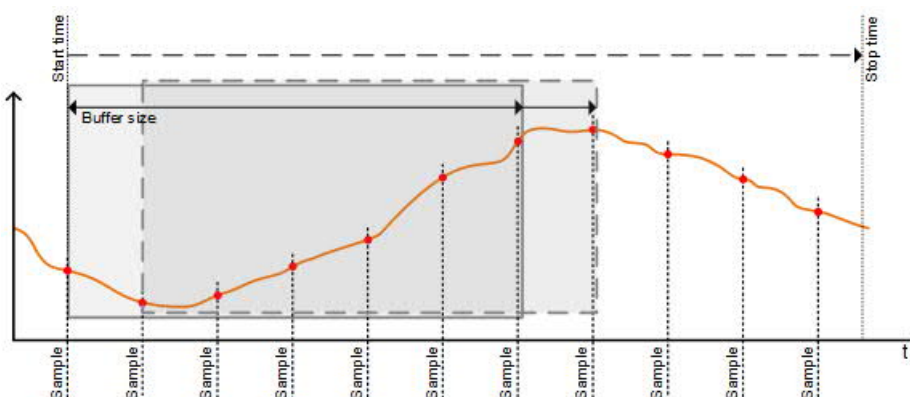
A BACnet client cannot reserve a Trend Log object. Every BACnet client can access the Trend Log object. In the case of access or modifications undertaken by several clients, the rule is that the most recent one always takes priority.

11.1 Trend functions

The trend log object supports the following functions.

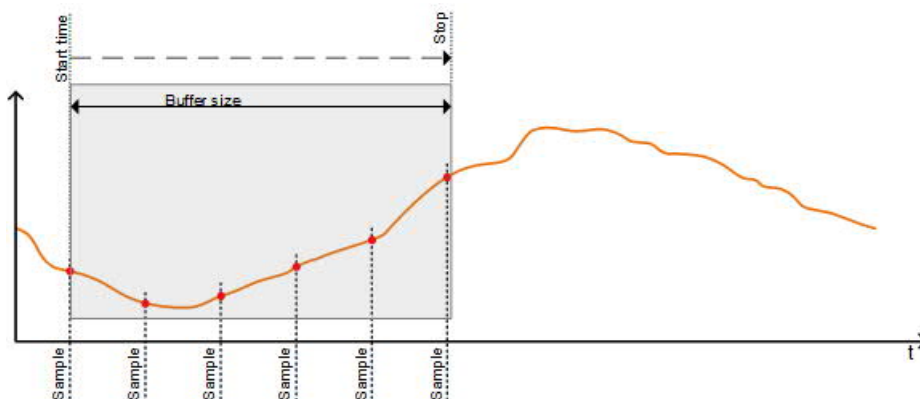
Continuous Run

The trend data is saved continuously (ring buffer). When the available memory area is full, the oldest data is overwritten by new data. You can define the Continuous Run function with the parameter Stop when full.



Single Run

The trend data is saved until the available memory area is full. You can define the buffer size [BufSize] within the range 2 to 5,000 entries. You can define the Single Run function with the parameter Stop when full.



Logging Type

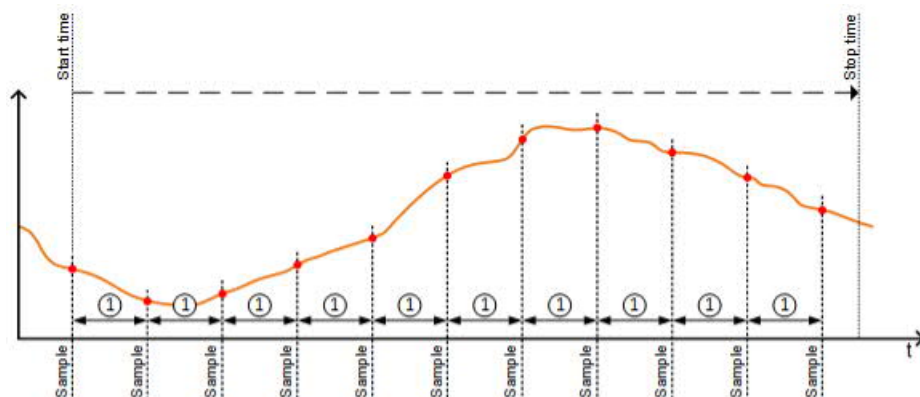
The parameter Logging Type [LogTyp] defines the logging type. The values are:

- POLLED: Periodic Sampling
- COV: COV Sampling
- TRIGGERED: Triggered Sampling

By setting the parameters accordingly, you can define combinations of Continuous Run / Single Run and Periodic Sampling / COV Sampling / Triggered Sampling.

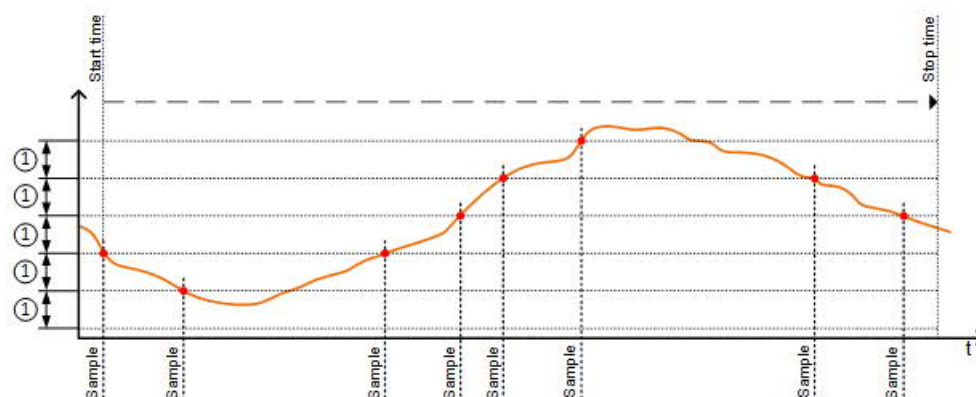
Periodic Sampling

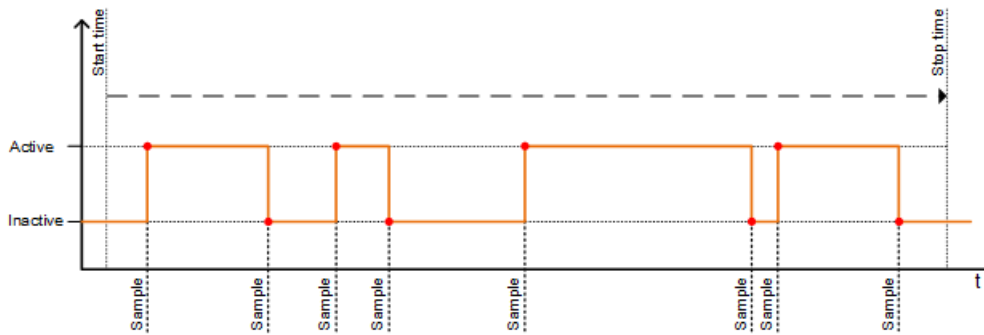
In Periodic Sampling data is acquired by sampling and storing values in a regular cycle. Periodic Sampling is supported by the trend log object and the trend log multiple object.



COV Sampling

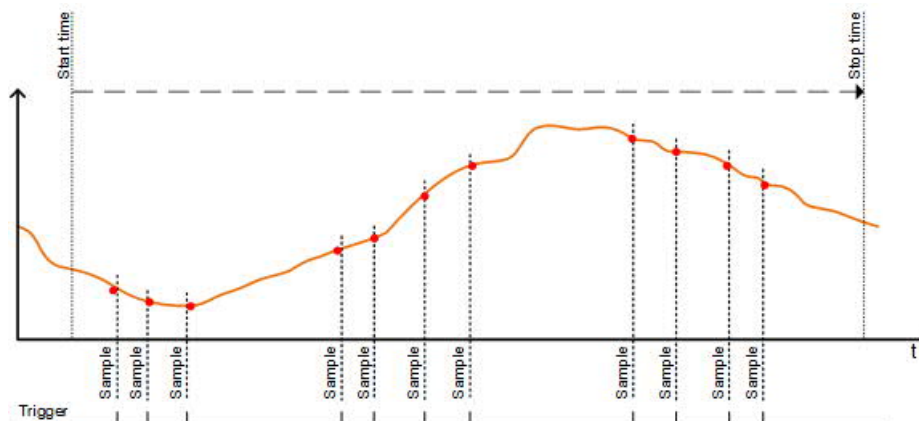
In COV Sampling data is stored based on a change of value (COV) of the referenced parameter. A COV subscription can be applied to all supported data types (analog, Boolean and multistate). The amount of change required to initiate a COV is set as a parameter in the object to be referenced. COV Sampling is supported only by the Trendlog object.





Triggered Sampling

In Triggered Sampling an application (e.g., via data flow interconnection) determines when values are acquired/logged and saved. Triggered Sampling is supported by the Trendlog object and the Trendlog Multiple object.



11.2 Editing parameters

Many of the parameters in the trend log object are definable only if:

- Enable for logging (Enable logging) [EnLog] is *inactive*
- The log buffer is *empty* (Record count = 1)
 - In this state, the following variables can be modified:
 - Start time [TiStt] and stop time [TiStp]
 - Interval [IV]
 - Buffer size [BufSize]
 - Record count [RecCnt] (can only be overwritten with 0: delete log buffer)
 - Notification threshold [NotifThd]
 - Input/Output address [IOAddr] (if an unavailable BACnet address is entered, an alarm is initiated)
- [EnLog] is *inactive* or *active*
- The log buffer is *not empty* (log count > 1)
 - In this state, only the following parameters can be configured:
 - Start time [TiStt] and stop time [TiStp]
 - Record count [RecCnt] (can only be overwritten with 0: delete log buffer)
 - Notification threshold [NotifThd]

The record count [RecCnt] can only be overwritten with 0. This deletes all the log data. After a write operation of 0, there is one entry showing the log status (record count = 1).

It is not possible to reload sampled data into the CFC Editor.

With a full loading procedure, any previously sampled data will be lost. With differential or delta loading the data will not be lost.

The PXM20 stores modified parameters in its internal memory cache. To display the data actually written, you must exit from the trend log object and re-select it.

11.3 Processing trend data in Desigo CC

The Trends application in Desigo CC lets you create and display online trends and offline trends. You can save, query, delete, edit and save trend data in Trend Views.

You can display the trend data in the Trend Viewer any time, even if Desigo CC is not connected to the site (no real-time data is available).

For more information, see *Desigo CC User Guide (A6V10415471)*.

12 Reports

You can create reports in Desigo CC about the functioning of the building automation and control system.

You can configure:

- The elements in the report (such as tables, plots, logos, form controls, text and so on), and their layout.
- Filters (such as name, condition, time, and/or row) to populate the elements of the report with information, e.g., if you want a report on a room's activity data over the past month, you could define a name filter and a time filter in an activities table.
- The formatting of the report elements and the page layout.
- The output type (PDF or XLS) and the output destination (file, email, or printer).

You can save your report definition for later use, run it, or schedule the report to be run at a specified time.

You can use reports as a reference or as a troubleshooting mechanism. Reports are helpful during system operation.

For example, you can:

- View a mixed report containing:
 - A table displaying details of all active events for a floor of a building
 - A table displaying a history report of events
 - A trends plot displaying the temperature variations gathered from temperature sensors
- Export trend data for statistical analysis to:
 - An XLS file
 - A CSV file (according to the EMC requirement)
- Schedule production of a report using macros and reactions
- Send a report by email, print it or save it as a PDF

You can export and import report definitions and logos.

You can also create and configure reports for operating procedures. These reports are used during assisted treatment to enter information about how the alarm or event is being handled.

13 Data storage

Large volumes of data are created in the Desigo system during engineering, commissioning and plant operation. The data is processed, saved, and archived as needed in accordance with type, generation, and meaning in the various system components.

13.1 Data categories

The application logic (control functions) and the required setting and configuration data are processed during engineering and loaded in the corresponding system products, such as Desigo CC or an automation station during commissioning.

This data is arranged according to two criteria:

- Data category
- Data ownership

Data categories: A common method of classifying data in building automation and control systems

| Data category | Description | Related terms |
|--------------------------|--|--|
| Program elements | Software components which perform a predefined task and have a known interface. | Function blocks, functions, program blocks, compounds, solutions |
| Setting parameters | Values that affect the program elements. | Setpoints, default values, limit values, address settings |
| Configuration parameters | Data in the form of defined constants, or data that influences the appearance or operability of the plant. | Description data, templates, profile, metadata |
| Process data | Physical process variables of the plant during operation. History or saved process data are plant data. | Measured values, state variables, calculated values |

Data ownership

Data ownership is based on the practical allocation of data to its owner. The owner, usually an organizational entity, a person, or a group of people, checks the data and is responsible for its scope and content. Data ownership shows in which Desigo system product the data is located and which tools are available for its management. Data ownership is divided in four groups.

The following table shows the four data ownership groups.

| Data ownership | Description | Owner | User |
|----------------|--|-----------------------------------|--------------------------|
| Program data | Software of a Desigo system product with basic data blocks. | Research & Development (R&D) (HQ) | Project/Library engineer |
| Libraries | Collection of predefined, specific, and tested program elements. Libraries can be copied and customized. | Library manager (HQ/RC) | Project/Library engineer |
| Project data | All data for the customer project or customer plant. | Contractor Project engineer | Plant operator |
| Plant data | Data from the customer plant saved permanently following commissioning. | Plant operator | Customer |

13.2 Program data

The executable software of a Desigo component is composed of programs. There are system and product components.

System components

System components are:

- System blocks for controlling the plant
- System interfaces which are implemented in every component and which control the data traffic between the components

Product components

Product components are the local subroutines responsible for the internal consistency of setup, startup, shutdown, navigation and display, etc. among the individual components.

Parameterization

There are two parameter types:

- Setting parameters: System and product components have predefined default settings. These values are application-independent, but always lie within the system limits.
- Configuration parameters: The system and product components have a basic configuration. The basic configuration of certain blocks is not complete and must be supplemented during engineering with, e.g., addresses of the I/O blocks.

The library or project engineers can adapt the setting and configuration parameters to the plant or project conditions.

13.3 Libraries

Libraries are needed during engineering. You can create loadable applications using libraries. Library elements are compiled from system basic components, e.g., the Desigo CC graphic library contains default graphics for visualizing plants during engineering.

You can copy and customize or extend libraries. Every engineering level has a library. Libraries can cover many combinations.

DXR2 automation stations are delivered with preloaded applications and only need to be configured.

There are three library types:

- HQ libraries are tested, well documented and delivered with the system version. Every new Desigo version contains new libraries.
- RC libraries are tested, well documented and contain country-specific characteristics. They are optional, independent or an addendum to the HQ library. Not all RCs offer comprehensive libraries.
- Project-specific libraries are not tested and documented.

Application libraries

Application libraries contain plant-specific functions (heating, cooling, control of electrical equipment, etc.) or templates for subsystem bindings. You can set up and manage libraries with the Xworks Plus (XWP) and Automation Building Tool (ABT) engineering tools.

PX libraries

The functional units of the PX application libraries are defined by compounds. You can copy, change and extend the compounds of the PX library.

Application libraries for PX are designed using the same application principles and are provided via Xworks Plus during project engineering.

PXC3/DXR2 libraries

The functional units of the PXC3/DXR2 application libraries are defined by application functions. You can copy, change and extend the application functions of the PXC3/DXR2 library.

Parameterization

The library elements have plant-specific or function-specific setting and configuration parameters:

- Setting parameters: The default values are defined by the application and usually do not require adjustment.
- Configuration parameters: The default values can be adapted as needed.

Graphics libraries

Graphics libraries contain graphics that represent the operating elements of the firmware and application libraries. The graphics are used in Desigo CC to visualize and operate plants. The elements of the graphics libraries depend on the elements of the application libraries. Any changes to the application libraries must therefore also be made to the graphics libraries.

The graphics libraries for Desigo PX and PXC3/DXR2 are identical.

13.4 Project data

There are three types of project data:

- Project data that is saved locally and then loaded into the system.
- Data on the Branch Office Server (BOS).
- Data that is loaded into the system with ABT Site and is not saved locally.

Project data is created during project engineering, when you create a project program using library components. Project programs define the sequence in which function blocks (programming elements) are processed, and what interconnections are used between blocks.

The library components are selected:

- In Xworks Plus (XWP) in the Solution Generator (recommended workflow for Desigo PX)
- In the CFC Editor from the compound and firmware libraries for Desigo PX
- In Xworks Plus and Automation Building Tool (ABT) from the Desigo room automation stations

Desigo room automation applications

Desigo room automation room applications are configured in ABT by selecting the required functions or model rooms.

RXB room applications

RXB room applications are configured in XWP by selecting the required functions or sample rooms. Data import in XWP to address the room devices is carried out in different tools, depending on the RX device.

| For device... | From tool... |
|-------------------------------------|---------------------|
| RXB and KNX/EIB third-party devices | PX KNX tool and ETS |
| LonWorks third-party devices | Standard LON tools |

PXC3 room applications

PXC3 room applications are programmed and configured in XWP/ABT by selecting and configuring the applications required for the rooms and room segments (application functions).

TX-I/O modules

The configuration of TX-I/O modules depends on the PX automation station:

- PX automation station with island bus connection: The XWP/ABT tools transfer the configuration data (IOC) to the target automation station PX/PXC3.
- PX automation station with P-bus connection: The configuration data (IOMD) from XWP is loaded into the I/O modules via the TX-I/O tool.

The IOC/IOMD configuration data is saved as project data.

Back up and restore

Project data is stored offline in XWP/ABT engineering tools. The backup and restore function creates a backup of the data and, in case of data loss, restores the data to the PX automation station.

Back up your data regularly to protect yourself against data loss.

The backup copy contains all necessary data of a PX automation station to ensure the automation station is fully functional after data restoration. You can back up and restore data on third-party automation stations. You can save engineering data on PX compact automation stations.

Backup

Data from the PX automation stations are saved as a backup copy on Desigo CC. The data is exported and saved as BACnet data.

To back up the data:

- The PX automation station must be connected and available (online).
- The PX automation station must support backup and restore.
- The building automation and control system must work smoothly.

Restore

You can restore data backed up on Desigo CC to the corresponding PX automation stations. The restored PX automation station automatically restarts after data restoration.

13.5 Plant data

Plant data is process data from the operation of a customer plant, that is permanently saved from the time of commissioning. Process data represents process variables in a building. The data is continuously changed by the environment, automation station and, in the event of physical outputs, the operator. Most process data is volatile. Few process data, such as adaptive control parameters and runtime totalization counts, remains available following a restart or power failure of the automation station. Process data can be archived as plant data.

13.6 Data transfer processes

Project data is transferred from the tools to the devices or other consumers online and offline via the system interface. PX automation stations are configured offline and then the data is downloaded onto the automation station. Desigo room automation stations (PXC3 and DXR2) can be engineered. DXR2 automation stations also have preloaded applications and only need to be configured.

Transfer operations

There are four transfer operations for data synchronization:

- Offline generation
- Online loading
- Online readback
- Offline import

Data transfer to PX automation stations

You can load a new program on a PX automation station while the old program is still running. The operation of the automation station is not interrupted. Process values remain intact. If you load a firmware on the automation station, you must restart the automation station.

Data transfer to Desigo room automation stations

If you load a new program on a Desigo room automation station, you must restart the automation station. The operation of the automation station is interrupted. Process values do not remain intact. If you load a firmware on the automation station, you must restart the automation station.

There are four loading units for Desigo room automation stations:

- Load program
- Load parameters (full download for DXR automation stations)
- Load configuration (settings on the automation station)
- Load firmware (programs, parameters and configuration are not lost)

Offline generation

Full code generation

Full code generation:

- Checks the overall application consistency (limits, identifiers)
- Converts the application into loadable units
- Generates the appropriate description data for configuration

You must compile the code to get the required performance. You cannot engineer a compiled program.

Delta generation

Delta generation (converts only the modifications):

- Improves performance
- Is faster than full code generation

Online loading

Full download

The full download transfers all loadable units into the automation station.

Delta download

The delta download:

- Copies additional blocks into the automation station
- Deletes blocks which are no longer valid
- Updates parameter settings

The delta download is faster than a full download. You do not need to interrupt the operation of the automation station.

The delta download helps prevent unintentional parameter changes. Online changed process data and settings parameters are protected against unintentional overwriting, provided the process data was not changed in the tool.

Download options

You can define what happens in the automation station before and after the download. You can define if:

- Parameters are read back before the download
- The program starts and/or the I/O bus is turned on after the download

Online readback

During readback of non-volatile process values and parameter settings the data in the automation station is aligned with the project data in the tool.

Readback comprises two steps:

1. Current parameter settings and non-volatile process values in the PX/PXC3 automation station are read from the automation station and copied to the data storage.
2. The values are updated in the CFC data storage or PXC3 program and configuration data storage.

The following data can be read back from PX/PXC3 to the project data storage in the XWP/ABT tools:

- Setting parameters changed in the PX/PXC3 automation station
- Changed, non-volatile process values or configuration data

Advantages

Reading back data has the following advantages:

- Outdated data in XWP/ABT is overwritten by current data and thus is available again for reloading programs in the PX/PXC3 automation station.
- During online changes of background variables (e.g., calendar), data between CFC and XWP/ABT and the automation station is retained.
- Current setting parameters and non-volatile process values (e.g., operating hours) are saved.
- The newest configuration is saved offline and can be used for, e.g., reports.

Runtime data cannot be read back. Only offline data can be read back. Only objects (not individual properties) can be read back. If a property is changed, the entire object (e.g., data point), that is, the last change on the object is read back. The last change per object is always valid.

Workflows for changes

There are two workflows for changes:

Workflow 1 (ideal workflow):

1. Perform readback before the changes
2. Perform changes offline
3. Download

Workflow 2:

1. Perform changes offline
2. Perform readback
3. Compile
4. Download

Offline import

You can import configuration and description data for the plant into Desigo CC. This is the same as the data downloaded to the automation station.

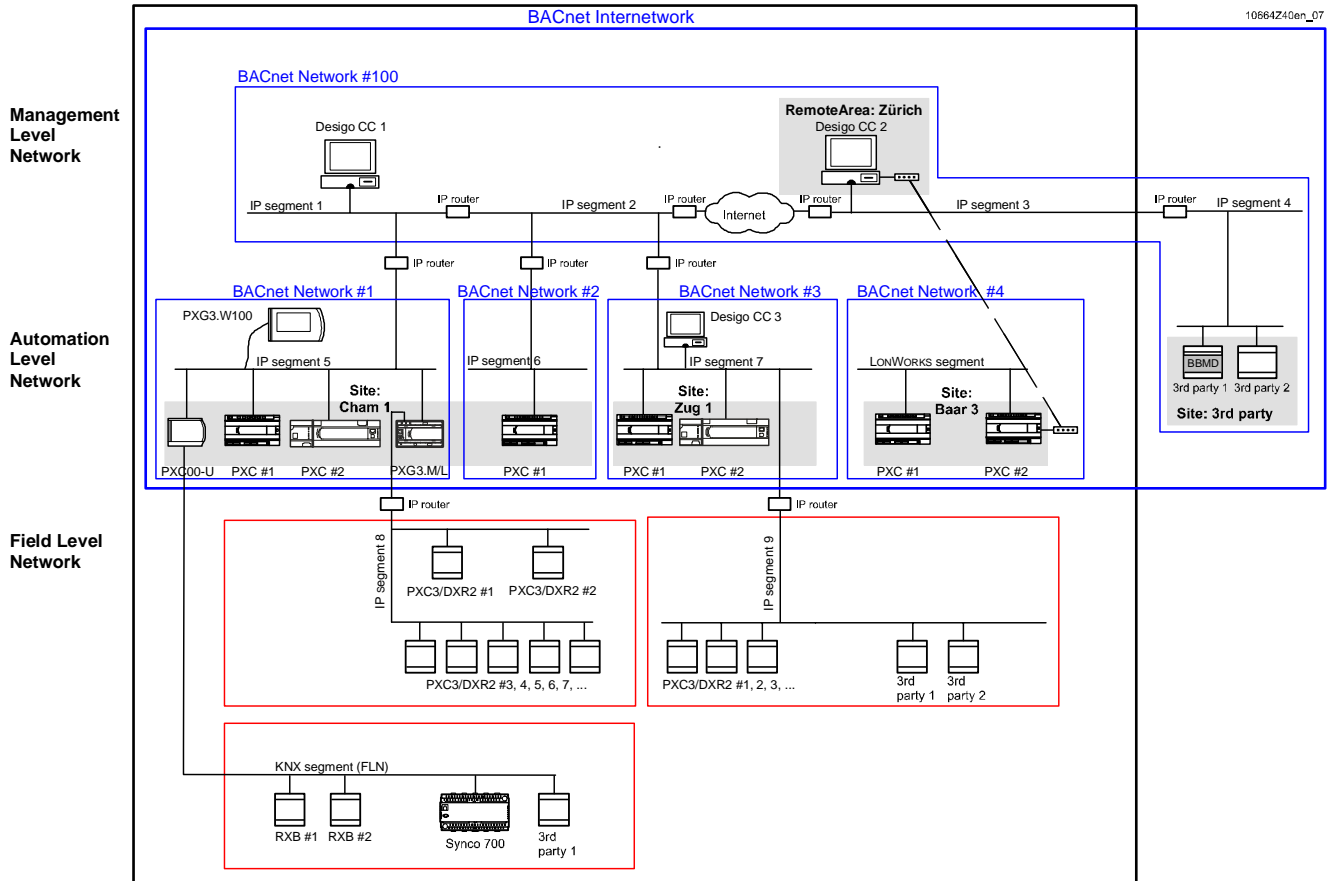
13.7 Texts

If you work with HQ or RC libraries, the texts are from a text database. These texts can be automatically translated, because they have a unique ID. Project-specific texts that are not from the text database cannot be automatically translated.

14 Network architecture

The Desigo system is divided into three network levels:

- Management Level Network (MLN)
- Automation Level Network (ALN)
- Field Level Network (FLN)

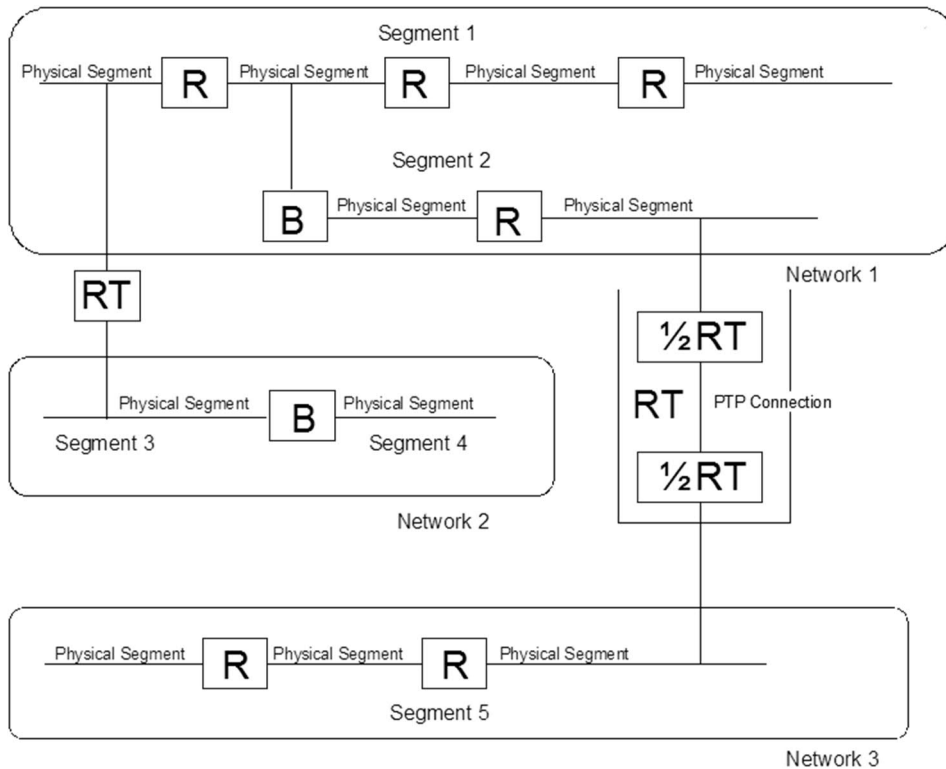


This classification is based on the functions performed at a given level, rather than on the communications protocol or medium. The MLN and ALN use the BACnet protocol. The transport protocol (Data Link Layer) is LonTalk or IP. The FLN uses LonWorks, KNX and MS/TP technology.

14.1 BACnet architecture (MLN & ALN)

Structuring

BACnet defines the following structuring of the network topology:



Key

- B Bridge, e.g., IP router, LonWorks router
- R Repeater, e.g., LonWorks physical repeater
- RT Router, e.g., PXG3
- ½RT Half router, e.g., PX..-T

Internetwork

In BACnet, the BACnet internetwork is defined as the largest BACnet unit. It consists of one or more BACnet networks. Only one active connection can exist between any two BACnet devices in a BACnet internetwork. All bus subscribers from the ALN and MLN, including BACnet third-party devices, are part of the BACnet internetwork. The devices in the FLN are part of the Desigo system but not part of the BACnet internetwork, because they do not communicate via BACnet.

Network

Desigo devices use LonTalk (BACnet/LonTalk), IP (BACnet/IP) or MS/TP (BACnet MS/TP) as their transport protocol. If different transport protocols are used, different physical networks are created, which must be connected to the PXG3 router. BACnet routers connect networks on the BACnet network layer and transmit messages via the network number.

If the transport protocol changes, the BACnet network also changes, e.g., BACnet devices that use LonTalk as their transport protocol are always located in a different network than devices that use IP as their transport protocol. This also applies to PTP connections.

Desigo devices use LonTalk (BACnet/LonTalk) or IP (BACnet/IP) as their transport protocol and MS/TP (BACnet MS/TP) to integrate third-party devices. If different transport protocols are used, different BACnet networks are automatically created, which must be connected to the PXG3 BACnet router. Multiple BACnet internetworks can be created on an IP segment by using different UDP port numbers.

Desigo establishes PTP connections only between operator units and a network. Operator units duplicate a virtual network since PTP connections demand a network at both ends.

Desigo CC does not use PTP.

Segment

Large networks are structured, that is, divided into several (logical) network segments for reasons of security, performance, size and (limited) address range of network devices. The segments must then be connected to routers of the corresponding transport protocol (e.g., LonWorks router, IP router).

In most cases it is not necessary to divide a BACnet/LonTalk network into several LonWorks segments (ALN). However, if it does prove necessary, it is not possible to use a LonWorks router, because this limits the length of the data packets. An L-switch (Loytec) can be used as a router on the ALN.

BACnet MS/TP networks cannot be segmented, because there are no associated routers.

With BACnet/IP some IP segments may be connected by IP routers. Since the IP router prevents broadcasting, the connection must be activated with the BACnet Broadcast Management Device (BBMD).

Physical segment

Physically, (cable) networks cannot be expanded as desired. Depending on electrical transmission properties and the data link layer based on it, repeaters must be added at specific cable lengths to amplify the signal. This divides the network into multiple, physical segments. A repeater does not impact the transmission protocol; it merely electrically connects two physical networks.

Dividing up the network into several physical segments may be necessary in LonWorks technology.

The physical segments are connected with physical or logical repeaters. Due to the limited buffer size of logical repeaters, only physical repeaters may be used on the ALN. Only one physical repeater may be located between any two nodes.

MS/TP is transmitted on a two-wire cable as per EIA-485/RS-485*. The length of the physical segment can be max. 1,200 m and can be extended with EIA-485 repeaters.

*TIA standard ([Telecommunications Industry Association](#)): TIA-485-A Electrical Characteristics of Generators and Receivers for Use in Balanced Digital Multipoint Systems (ANSI/TIA/EIA-485-A-98) (R2003)

Desigo site

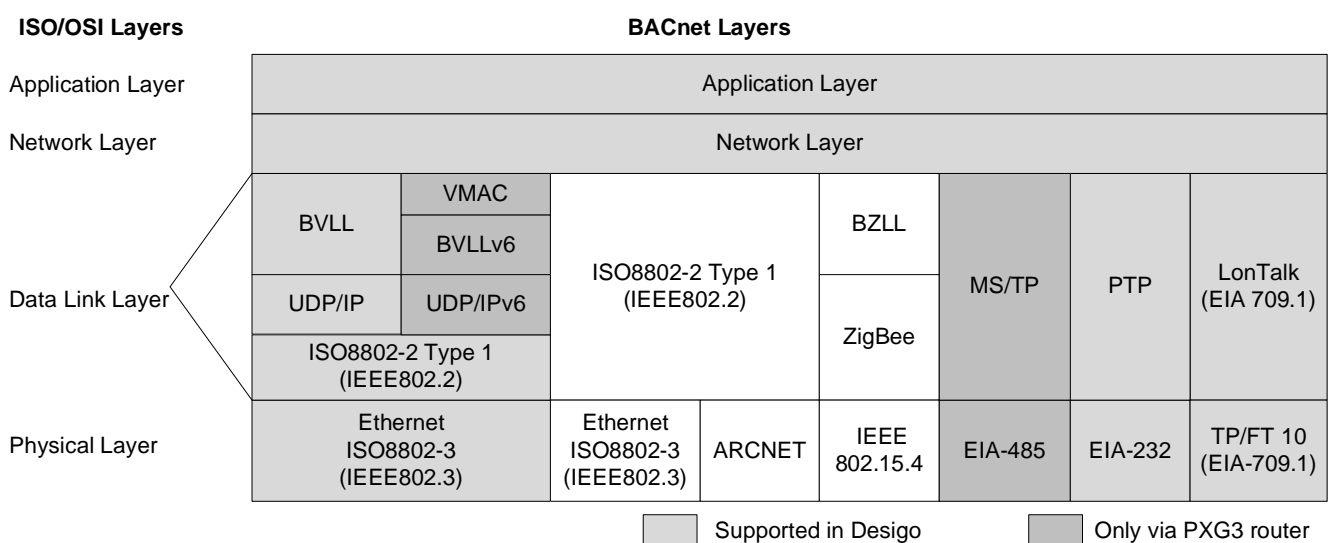
A site is an independent and self-contained logical entity within the building automation and control system. This type of structuring is not defined by BACnet, and is therefore largely independent of the BACnet network topology. The BACnet devices bound to a site can therefore be placed anywhere within a BACnet internetwork. A site cannot extend across a PTP connection. Communication occurs only within the site, but data can be exchanged with any device on the BACnet internetwork.

Only automation stations (PXC/PXC3) and LonWorks system controllers (PXC...D+PXX-L11/12)) are assigned to the sites, by special structuring of the Device ID and Device Name. These products are considered third-party devices for the purposes of a site.

Protocol layer model

Desigo supports:

- BACnet/IP
- BACnet/LonTalk (LonWorks technology)
- BACnet/PTP
- BACnet MS/TP
- BACnet/IPv6 (only via PXG3 BACnet router)



BACnet directly over ethernet, ZigBee or ARCnet is not supported.

Application layer

The BACnet application layer defines services, objects and their characteristics. From the network viewpoint, the Device object is important. The object ID and the object name must be unique within the BACnet internetwork.

| | | | |
|-------------------|----|-----|------|
| Application Layer | | | |
| Network Layer | | | |
| LonTalk | IP | PTP | MSTP |

Device ID

The Device ID is the object ID of the BACnet device object.

The device ID is divided into the following categories:

| Category | Device ID | | | Description | |
|--------------|-------------|-----------------|---------------------------------------|----------------------|---|
| | Object type | Object instance | | | |
| | | 2 bit | 10 bit | 10 bit | |
| Unconfigured | DEVICE | 00 | 0000000000 0000000000 | | All unconfigured devices |
| Class A | DEVICE | 00 | xxxxxxxxxx xxxxxxxxxxxx (1...1048576) | | Third-party devices |
| Class B1 | DEVICE | 01 | | Device number | Stationary operator units (Desigo CC) |
| | | | 0000000000 | xxxxxxxxxx (1...999) | |
| Class B2 | DEVICE | 01 | | Device number | Mobile operator units / tools (PXM30/40/50-E) |
| | | | 0000000001 | xxxxxxxxxx (1...999) | |
| Class B3 | DEVICE | 01 | | Device number | System devices (BACnet router) |
| | | | 0000000010 | xxxxxxxxxx (1...999) | |
| Class C | DEVICE | 10 | Site number | Device number | Automation stations (PXC...) |
| | | | xxxxxxxxxx (1...999) | xxxxxxxxxx (1...999) | System controllers (PXC...) |
| Class D | DEVICE | 11 | xxxxxxxxxx xxxxxxxxxxxx (1...1048576) | | Reserved |

Device name

The device name is the object name of the BACnet device object.

Guidelines

Different rules for object names apply for configuring TD (Technical Designation), UD (User Designation), or FD (Free Designation):

- The TD is generated from predefined partial names, separated by an apostrophe ('), that show the technical hierarchy with plant, partial plant, and component. The TD is supplemented by site name and pin name.
- The names may consist of upper- and lowercase letters and numbers 0 to 9. The site name is separated by a colon (:) and the pin name by a period (.). The maximum total length is 30 characters.
- The UD is formed similar to the TD based on partial names. However, users determine the partial names, structure, and separators. The names consist of upper- and lowercase letters, numbers 0 to 9, and separators, such as _;=, etc. The maximum total length (including site name and pin name) is 80 characters.
- The FD is a freely assigned name consisting of letters, numbers and a few special characters, limited within the system only by uniqueness and length. The maximum total length is 80 characters, ten of which, plus one separator, are reserved for the site name.

| Category | Device name | Description |
|--------------|-------------------------------------|---|
| Unconfigured | "" | Empty string for unconfigured devices |
| Class A | | No rules |
| Class B1 | | Meaningful text for the user (this text is used as a reference for the alarm recipient) |
| Class B2 | Model name + device ID | Model name + " " + 8 character device ID (hexadecimal). The device name for temporary devices is generated automatically. |
| Class B3 | | Max. 25 characters |
| Class C | Site name + automation station name | Site name + ":" + automation station name |

Examples from the topology at the beginning of the chapter:

| Category | Site number | Device number | Device ID | Site name | Device name | Device name |
|----------|-------------|---------------|------------|-----------|---------------|------------------|
| A | – | 1 | 0x02000001 | – | Third-party 1 | Third-party 1 |
| B1 | – | 1 | 0x02100001 | – | Desigo CC 1 | Desigo CC 1 |
| B2 | – | 15 | 0x02100401 | – | – | PXM20TMP0210040f |
| C | 1 | 1 | 0x02200401 | Cham | PXC #1 | Cham:PXC #1 |
| C | 1 | 2 | 0x02200402 | Cham | PXC #2 | Cham:PXC #2 |
| C | 1 | 3 | 0x02200403 | Cham | PXC... #1 | Cham:PXC... #1 |
| C | 2 | 1 | 0x02200801 | Zug | PXC #1 | Zug:PXC #1 |
| C | 2 | 2 | 0x02200802 | Zug | PXC... #1 | Zug:PXC... #1 |
| C | 3 | 1 | 0x02200C02 | Baar | PXC #1 | Baar:PXC #1 |

BACnet device parameters

The BACnet device parameters are written to the devices during commissioning. These parameters include the following values:

| Designation | Description |
|--------------------------------|---|
| Max APDU Length Accepted | Maximum length of application message (Application Protocol Data Unit) supported for this device. The length depends on the transport medium used, and the capacity of the device buffer. The length of the APDU must always be less than the length of the smallest NPDU (Network Protocol Data Unit) between the different bus subscribers. |
| Beispiel | There are two IP networks linked by a PTP connection. The two IP bus subscribers could have a maximum APDU length of 1476 octets. However, since the maximum NPDU length of the PTP connection is 500 octets, the maximum APDU length of both devices must be set to 480 octets. |
| Values for LonTalk | Range: 50/128/206 Octets |
| | Default value: 206 Octets |
| Values for MS/TP | Range: 50/128/206/480 Octets |
| | Default value: 480 Octets |
| Values for IP (equal for IPv6) | Range: 50/128/206/480/1024/1476 Octets |
| | Default value: 1476 Octets |
| APDU Segment Timeout | Timeout of an APDU segment (= part of an APDU). This value must be identical throughout the internetwork. |
| | Range: 1000...5000 ms |
| | Default value: 2000 ms |
| APDU Timeout | Timeout for an acknowledged message. This value must be identical throughout the internetwork. |
| | Range: 1000...5000 ms |

| Designation | Description | |
|------------------------|--|---------|
| | Default value: | 3000 ms |
| Number of APDU Retries | Number of retries in the event of an APDU or APDU segment timeout. This value must be identical throughout the internetwork. | |
| | Range: | 1...5 |
| | Default value: | 3 |

Window size

To transfer large data packs, BACnet uses the windowing algorithm. Windowing means that instead of acknowledging individual segments separately, the acknowledgement applies to a specific number of segments, referred to as a window.

Definitions for Desigo

The window size is permanently set to four for all Desigo devices, so that for segmented messages, only every fourth segment is acknowledged.

Network layer

The most important information in the network layer is the network number of the BACnet network.

| | | | |
|-------------------|----|-----|------|
| Application Layer | | | |
| Network Layer | | | |
| LonTalk | IP | PTP | MSTP |

Network number

The network number is the unique identification of the BACnet network. There are stationary and temporary networks:

- Stationary networks are defined during commissioning and then remain unchanged.
- Temporary networks are created when a tool (e.g., XWP/ABT) dials into a network via PTP.

| Range/Value | Description | |
|---------------|--|-------------|
| 0 | Reserved for applications with only one BACnet network in a BACnet internetwork, that is, where there are no BACnet routers. | |
| 1...65280 | Network number for stationary BACnet networks. You can select any network number in this range. We recommend that you form categories, e.g.: | |
| | BACnet/LonTalk networks via (half)router: | 1...99 |
| | BACnet/IP network (common network): | 100 |
| | Desigo CC or XWP/ABT connected via BACnet/PTP ¹ | 1000...1099 |
| 65281...65534 | Reserved for temporary BACnet networks. Not yet used in Desigo. | |

Router parameters

The router parameters are written to the BACnet router during commissioning. The following information is required for each port (logical connection to network):

| Designation | Description | |
|---------------------|---|------------|
| Network Number | Network number of the directly connected network. | |
| Max NPDU Length | Max. message length supported in this network. This value depends on the transport medium used. | |
| Values for LonTalk: | Range: | 50/228 |
| | Default: | 228 |
| Values for MS/TP: | Range: | 50/228/501 |
| | Default: | 501 |

| Designation | Description | |
|--------------------------------|-------------|--|
| Value for IP (equal for IPv6): | Range: | 50/228/501/1497 |
| | Default: | 1497 |
| Values for PTP: | Range: | 50/228/501 |
| | Default: | 228 for LonTalk / 501 for Ethernet/IP |

Hop counter

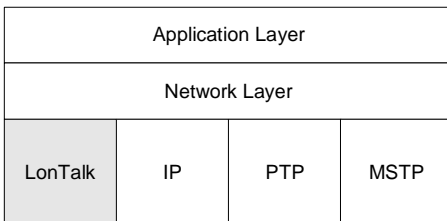
Every BACnet that is routed to another BACnet network has a hop counter. The counter reading is reduced by one with each pass of the BACnet router. When the counter reads 0, the message will not be routed further. This prevents continuously circulating messages.

Definitions for Desigo

For Desigo the hop counter is initialized with a fixed value of 5. This means that a message can pass through a maximum of four BACnet routers.

LonTalk data link layer

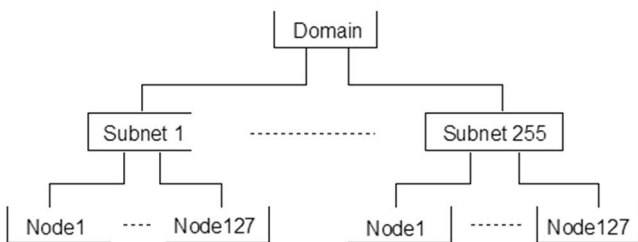
The LonTalk data link layer is supported by the PX automation station and by the operator units and tools. LonTalk ist he communications protocol for LonWorks technology.



Addressing under LonWorks technology

Physical address, neuron ID: The Neuron ID is the physical address for a LonWorks device. It is a unique 48-bit (6 byte) identifier which is assigned to each neuron chip during manufacturing.

Logical address: The logical LonTalk address is written to the LonWorks node during commissioning on the network side.



Domain ID: The domain ID is the highest unit in the LonWorks addressing system. Data can only be exchanged within a domain. A gateway is required for inter-domain communication. The domain ID can be 0, 1, 3 or 6 octets in length. A domain can consist of up to 255 subnets.

Subnet ID: The subnet is a logical collection of up to 127 nodes within a domain. The bus traffic within a subnet can be kept local by using BACnet routers. Subnets must never be defined across a router.

Node ID: Unique identifier within the subnet. Each node can be addressed uniquely within a domain by the subnet ID and the node ID.

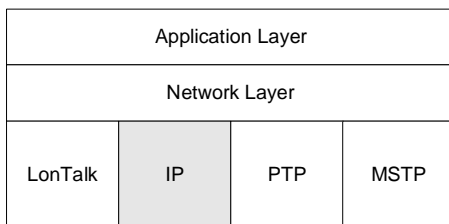
Group ID: The group address is a type of addressing. The group address is not used in BACnet.

On the ALN, the following rules apply to Desigo:

| Designation | Values/Range | Description |
|-------------|--------------|---|
| Domain ID | 0x49h (73) | The default length of the domain ID is one octet and the default value is 0x49h (73). |
| Subnet ID | 1...255 | The subnet ID is a consecutive number that starts with one. The subnet ID is incremented by one when a subnet is full (no free node IDs). |
| Node ID | 1...100 | This range is for automation stations (PXC), system controllers (PXC...) and system devices (BACnet routers). |
| | 101...120 | Operator units and Desigo CC are assigned to this range. |
| | 121...127 | Temporary operator units and tools (XWP/ABT) look for a free node ID in this range. |

IP data link layer

An additional layer, the BACnet Virtual Link Layer (BVLL), is used for BACnet over IP. This layer transmits broadcast messages across IP routers.



Below the BVLL, BACnet relies on UDP (User Datagram Protocol). Unlike TCP (Transmission Control Protocol), UDP supports broadcast messages. The connection monitoring (carried out by TCP) is resolved in the Application Layer.

All media, such as ethernet, are available if supported by IP as physical layers.

For detailed information on the IPv6 data link layer, see *Ethernet, TCP/IP, MS/TP and BACnet basics* (CM110666).

IP addresses

The IP address of stationary and temporary operator units can be set automatically via DHCP (Dynamic Host Configuration Protocol) provided that there is a DHCP server in the network. The use of DHCP is not recommended with automation stations and BACnet routers. DHCPv6 is currently not supported for IPv6.

DHCP is not allowed for devices using integrated BBMD functionality.

The IP addresses must be agreed upon with the IT department.

RFC1918 defines three specific address areas for private networks. IP addresses within these ranges are not routed:

10.0.0.0 - 10.255.255.255 Subnet mask: 255.0.0.0

172.16.0.0 - 172.31.255.255 Subnet mask: 255.240.0.0

192.168.0.0 - 192.168.255.255 Subnet mask: 255.255.0.0

For IPv6, IP addresses and private address ranges are defined differently. See *Ethernet, TCP/IP, MS/TP and BACnet basics* (CM110666).

IP address: Host address of the network subscriber.

Subnet mask: Subnet mask of the IP segment in which the device is located. This value must be aligned with the other IP devices.

The subnet mask is required for the identification of broadcast messages and for communication across IP segments. The subnet mask and target IP address enable the transmitting IP device to decide whether the data packet can be delivered directly to the target device or if it must be forwarded via the default gateway.

For IPv6, the subnet mask corresponds to the network prefix. See *Ethernet, TCP/IP, MS/TP and BACnet basics* (CM110666).

Default gateway: IP address of the IP router. This value is relevant for communication across IP segments.

UDP port number

For BACnet/IP to use UDP, a UDP port number must be defined. Only devices with the same UDP port number can communicate with each other.

Port numbers are divided into the following classes by the IANA (Internet Assigned Numbers Authority):

- Well Known Port Numbers: Fixed port numbers assigned by IANA (0... 1023)
- Registered Port Numbers: Numbers registered with IANA (1024...48151)
- Dynamic and/or Private Ports Dynamically assigned or privately used port numbers (49152...65535)

For BACnet, port number 47808 (0xBAC0) is registered with IANA.

If there are several BACnet internetworks on an IP network, they can be separated by different port numbers. Using several internetworks can be helpful in very large projects, for migration, and to encapsulate sections of a plant with different reliability criteria. Since Desigo CC communicates simultaneously with multiple internetworks, the operation is not restricted.

However, only one port number is registered for BACnet with the IANA. If additional UDP port numbers are required, we recommend the use of port numbers 47809 to 47823 (0xBAC1...0xBACF). This does not comply with IANA regulations. This range is reserved for future applications and should not be used. There is only a very small chance that these ports might be used elsewhere. To avoid clashes, do not use any port numbers from the range of dynamic or private ports. See www.iana.org/assignments/port-numbers.

BACnet Broadcast Management Device (BBMD)

The BBMD is required as soon as IP routers are used in a BACnet network. IP routers limit broadcast messages to the local IP segment, that is, they do not allow any broadcast messages to pass through. In order to distribute BACnet broadcast messages across IP segments irrespective of this limitation, a BBMD is required in the relevant IP segments. If a BBMD receives a broadcast message, e.g., within the local IP segment, it transmits this as a unicast message to all other BBMDs. The BBMDs then transmit the received message to their own local IP segments. BACnet refers to this as two-hop distribution:

1. Hop: BBMD sends a unicast message to all other BBMDs.
2. Hop: They then distribute the message to all BACnet devices in the local IP segment.

One-hop distribution can be implemented with Direct Broadcasts. In this case the BBMD sends a direct broadcast to all remote IP segments. This broadcast is received by all IP bus subscribers in the relevant segment. Not all IP routers support Direct Broadcasts.

IPv6 (BVLv6) only supports two-hop BBMD. Broadcasts are implemented via IPv6 multicasts. See *Ethernet, TCP/IP, MS/TP and BACnet basics* (CM110666).

BBMDs ensure that broadcast messages are distributed in a BACnet network. They are grouped by BACnet network. A maximum of one BBMD is allowed in any one IP segment.

BACnet network #100 is separated by IP routers. The Internet also contains IP routers. This is why different segments are shown before and after the Internet cloud. BBMDs are required so that BACnet broadcast messages are available in all IP segments.

BBMD parameters

The BBMD parameters are written to the BBMD or (for Desigo) to the BACnet router during commissioning. The following information is required for each BBMD in the BACnet network:

| Designation | Description |
|----------------|---|
| IP address | IP address of the BBMD. |
| UDP port | UDP port number of the BBMD. |
| Broadcast mask | If the BBMD is to be addressed via direct broadcast (one-hop distribution) the subnet mask of the BBMD must be specified. Since not all IP routers support this mechanism, direct broadcasts are not supported by default. Two-hop-distribution is always possible. The broadcast mask is then 255.255.255.255. Not required for IPv6. |

Foreign device

A foreign device is a (remote) BACnet device in a remote IP segment. It registers with a BBMD in order to send or receive broadcast messages. Registration with a BBMD involves making an entry in its Foreign Device Table (FDT). The registration must be renewed at regular intervals.

The foreign device does not send broadcast messages, but passes them as unicast messages to the BBMD for distribution. The BBMD in turn passes incoming broadcast messages as unicast messages to the foreign devices in its FDT.

In the Desigo system, Desigo CC, XWP/ABT, and PXM30/40/50-E can be operated as foreign devices. IPv6 does not support foreign devices

Examples from the Desigo topology

- IP Segment 1: Desigo CC 1 does not have to be configured as a foreign device, because this IP segment contains a BBMD.
- IP Segment 2: Desigo CC 3 does not have to be configured as foreign device, because this IP segment contains a BBMD.
- IP Segment 3: This segment only contains Desigo CC 2. To enable Desigo CC 2 to receive and send broadcast messages, it must register with a BBMD as a foreign device. It does not matter with which BBMD it registers.

Foreign device parameters

If a BACnet device operates as a foreign device, the IP address and UDP port number of the BBMD must be specified.

| Designation | Description |
|--------------------|---|
| IP Address of BBMD | IP address of the BBMD with which the foreign device registers. |
| UDP Port of BBMD | UDP port number of the BBMD with which the foreign device is registered. The default is 0xBAC0. |

The recording interval (Time-To-Live) for Desigo products is set at 300 seconds (= 5 minutes).

PTP data link layer

The PTP Data Link Layer is used for remote management over the telephone line. Unlike LonTalk and IP, PTP does not allow the creation of a network. The PTP connection is always between two half routers, and between two different BACnet networks. Several BACnet networks may be located at each end of the PTP connection. Only one active communication line can exist between any two BACnet networks or between any BACnet devices.

The half-router function is implemented in Desigo CC, XWP/ABT and PX.

| | | | |
|-------------------|----|-----|------|
| Application Layer | | | |
| Network Layer | | | |
| LonTalk | IP | PTP | MSTP |

PTP connections are only possible between Desigo CC, XWP/ABT und PX. PTP connections between PXs are not permitted.

PX devices which can be reached via PTP always belong to a separate site. With reference to the topology at the beginning of the chapter, the site named Baar must not be combined with the sites named Zug or Cham. Several PXs per site can be used as half routers. When establishing a communication, the best-performing connection is always selected. Redundancy is not allowed, that is, several simultaneously active connections in a given BACnet network are not allowed.

With Desigo CC, a separate, independent, internal BACnet internetwork is created for each Data Link Layer type. Routing between LonTalk, IP or PTP is therefore not possible.

The following parameters are required for each half router:

| Designation | Description |
|----------------------|--|
| Local network number | The BACnet network number to which the half router belongs. With PX, the local network number is the same as the device's own network number. Desigo CC supports all three different Data Link Layers (IP, LonTalk and PTP). They are handled internally as independent BACnet internetworks. This means that routing between the different Data Link Layers is not possible. Therefore, the local network number can be allocated to IP and/or LonTalk independently of the networks. However the local network number must be unique among all networks which could possibly be reached from Desigo CC via a PTP connection. Recommendation: If Desigo CC has an additional Data Link Layer (IP and/or LonTalk), the local network number of this network should be used (example in the beginning of the chapter: For Desigo CC 2, the local network number of BACnet network #4 should be adopted). In Desigo CC with only one PTP connection, the local network number must be in the range 1000 to 1099. (Example: Desigo CC 3 -> #1000). |
| COM parameter | For the PX half router, the COM port to which a modem or null modem is connected must be specified. |
| Modem parameters | The modem parameters contain individual settings for the relevant modem types. Predefined parameter sets are available for the PX half router. |

The following parameters are required for each PTP connection starting from a PX half router:

| Designation | Description |
|-----------------------|---|
| Remote network number | This network number determines the BACnet network in which the remote partner device is located. In the Desigo system this is the local network number of Desigo CC. |
| Remote area name | The remote area name stands for a peer-to-peer remote network number of the network containing Desigo CC. During configuration, the remote area number lets you assign a clear name to the (remote) alarm recipient rather than a network number. |
| Telephone number | The telephone number for access to the remote device. |
| Performance index | The performance index refers to the quality of the router data connection. If multiple PX half-routers are available in a PX site, and if a connection to a remote network is to be established, the router with the best performance index is selected. If no connection is established, the router with the next best performance index automatically tries to connect. |
| | Range: 0...255 |

For each PTP connection in Desigo CC, only the telephone number needs to be defined.

Data link layer MS/TP

Data Link Layer Master/Slave Token Passing MS/TP is another protocol variant for BACnet. Desigo supports this variant via a specific router that connects BACnet MS/TP to BACnet/IP.

MS/TP is based on the physical layer EIA-485/RS-485 and supports baud rates up to 76.8 kbps. Up to 256 devices can be connected to one MS/TP segment (in theory, dependent on their unit load).

| | | | |
|-------------------|----|-----|------|
| Application Layer | | | |
| Network Layer | | | |
| LonTalk | IP | PTP | MSTP |

Addressing MS/TP devices

Each device has its own unique MAC address. The MAC address is one octet long and defined as follows:

- 0-127 reserved for master devices
- 0-254 reserved for slave devices
- 255 reserved for broadcasts

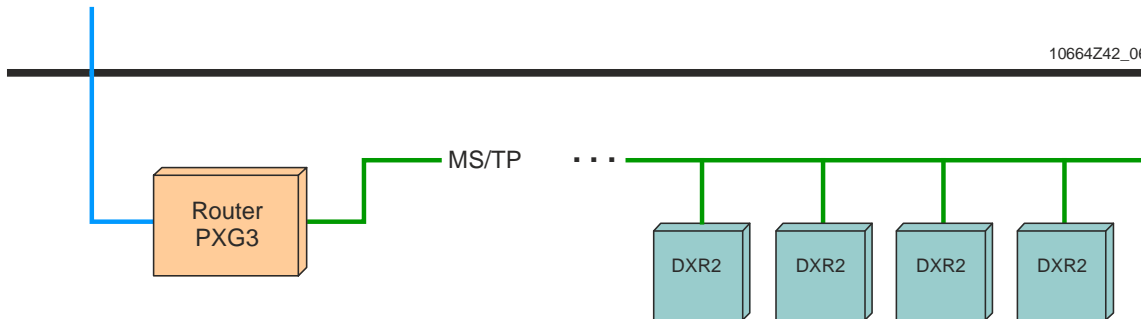
The MAC address can be set via DIP switch (hardware) or related configuration tools (software) for each device.

Structuring

MS/TP is transmitted via two-core cables as per EIA-485/RS-485. The maximum length of a segment is 1200 meters. 64 devices are allowed on a segment.

Different segments can be interconnected via repeaters to form a larger EIA-485-network. The specific electrical properties, such as polarity, common signal ground, terminating resistances, etc. must be taken into account. The actual, possible network size and maximum transmission rate primarily depend on the network structure. Establishing a network in daisy chain form is best.

BACnet/IP



Due to relatively difficult, electrical conditions imposed by EIA-485 wiring and limited data transmission capacity, we recommend using BACnet MS/TP only for devices with low data volumes that are geographically far apart.

For devices with larger data volumes and shorter distances to the Desigo automation station, integration in Desigo primarily should be carried out via TX Open or PX Open.

System devices

PXG3 is a BACnet router that routes BACnet telegrams between BACnet networks and different data link layers. It is available in two versions:

- PXG3.L: (triangle router) Simultaneous routing between Ethernet/IP, LonTalk, and MS/TP
- PXG3.M: Routing between Ethernet/IP and MS/TP

See *BACnet router for BACnet/Ethernet/IP, BACnet/LonTalk, BACnet MS/TP PXG3.L, PXG3.M (CM1N9270)*.

An individual BACnet IPv6 data link can be used as an option for the router. As a result, the PXG3.M is turned into a triangle router and the PXG3.L to a square router. The router can be configured either via XWP or the integrated web server.

BACnet address

Every BACnet device in the BACnet internetwork can be accessed via its BACnet address.

The BACnet address is defined by the BACnet standard and comprises the following elements:

| Designation | Description | |
|--------------------|---|---|
| Network number | Network number of the BACnet network in which the device is located. The network number is only parameterized on devices with BACnet router functionality (including half router) and is implicitly valid for all BACnet devices in the BACnet network. | |
| BACnet MAC address | Address specific to the transport protocol. This address is written to the device in the commissioning phase. BACnet MAC address for: | |
| | LonTalk: | 2 octets, subnet ID und node ID |
| | IP: | 6 octets, IP address and UDP port |
| | IPv6: | 3 octets as virtual MAC address (VMAC) |
| | MS/TP: | 1 octet, MAC address (master 0-127, slave 0-254, broadcast 255) |

BACnet device address

Each BACnet device has a device address. This address is written to the device during the network commissioning process. The BACnet device address is unique within the BACnet internetwork. The term BACnet device address is an in-house term rather than an official BACnet term.

| Designation | Description |
|-------------|--|
| Device ID | Object identifier of the BACnet device object. The device ID is unique within the BACnet internetwork. |
| Device name | The object name of the BACnet device object. The device name is unique within the BACnet internetwork. |

For information about the direct addressing of BACnet references to objects in other networks and media, see *Desigo Ethernet, TCP/IP, MS/TP and BACnet (CM110666)*.

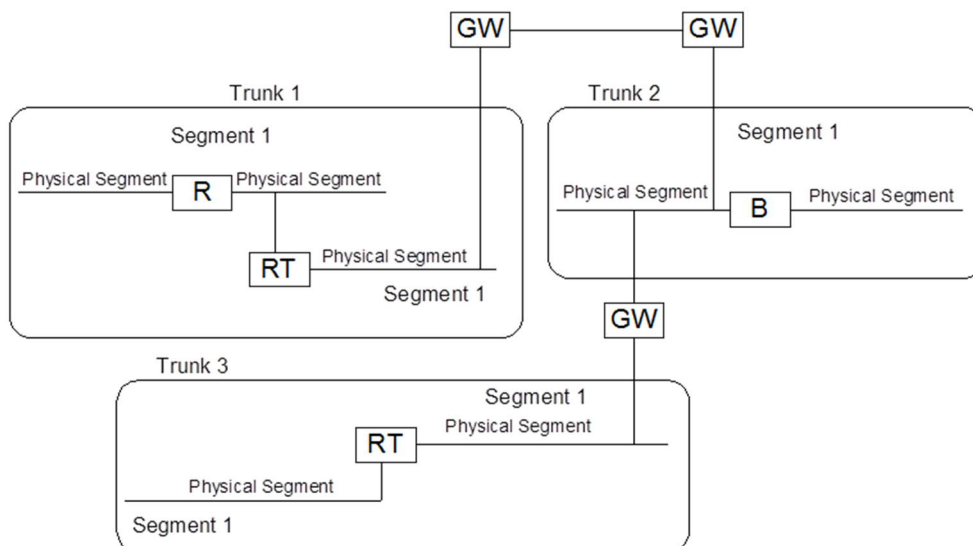
14.2 LonWorks architecture (ALN)

With the LonWorks-based communication protocol complete networks made up of interoperable products can be created. The protocol conforms to ISO/IEC 14908 (worldwide), EN 14908 (Europe), ANSI/CEA-709/852 (U.S.) and is also standardized in China. See www.lonmark.org.

LonWorks is suited for use with different types of transmission media, such as twisted pair cables, power line, RF, fiber optics or IP (TCP/IP and UDP/IP) It supports straightforward installation with different cabling topologies (e.g., star or line). The connection of objects via bindings (e.g., standard network variables (SNVTs), standard configuration properties (SCPTs)) can be defined at the project engineering stage or can be adapted in the field.

Structure

The following figure shows the structure of a Lon network in the FLN



Key

- R Repeater, e.g., LonWorks physical repeater
- B Bridge, e.g., L-Switch (Loytec)
- RT Router, e.g., LonWoks router
- GW Gateway, e.g., PXC..., RXZ03.1

See *LonWorks networks Checklist (CA110335)*.

Trunk

A trunk holds all devices that can communicate with each other directly or via repeater, bridge or router. The term trunk is specific to the Desigo system. One trunk corresponds to one LonWorks project. Trunks can be connected via gateways.

Segment

A trunk can be divided into segments. The segments are connected by a router. If there are no routers, the trunk comprises only one segment.

Physical segment

The physical segment is the communication medium. LonWorks devices are connected to the physical segment. One segment can be divided by bridges or repeaters into several segments. The number of devices per physical segment is limited.

System devices

Gateway

The gateway links trunks. It operates on the application layer of the ISO/OSI layer model. The following LonWorks gateways are available:

The RXZ03.1 point coupler provides a fixed number and type of LonTalk network variables (NV). Each side of the point coupler belongs to a trunk or LonWorks project. The point coupler can be used to implement time-critical connections between two trunks. The point coupler integrates third-party devices that have been engineered with a different tool.

Loytec L-Proxy and Sysmic XFM-LL are freely programmable point couplers. The XFM-LL device may be used, when depicted like a standard third-party device (configuration via its own tool).

The PXX-L.. extension modules let you connect LonWorks devices to the PXC..D modular series.

Router

The LonWorks router operates on the network layer of the LonWorks protocol. It filters data packets based on their subnet ID or group ID. Subnets or groups must never be defined across a router, that is, the subnet IDs and group IDs at each end of the router must never be the same. Routers are used where there is heavy local network traffic. They allow the unloading of unaffected devices from the network traffic. In Desigo there are no large LonWorks networks, as the FLN is divided into trunks. Routers are only required in exception cases.

L-Switch (Loytec)

The L-Switch filters the package on the basis of the subnet/node ID or group ID. It automatically learns the topology and forwards the data packets accordingly. The L-switch does not have to be configured. Unlike the router, there is no need to take account of any addressing limits (allocation of Subnet ID or Group ID).

Physical repeater

LonWorks has physical and logical repeaters. The physical LonWorks repeater does not filter the data packets. It regenerates the electrical signal. One physical LonWorks repeater can be used in the path between any two devices within a segment.

In logical repeaters, the data packet is processed by the neuron chip. This enables several logical repeaters to be connected in series. The disadvantage is that the logical repeater must be configured, and that owing to the limited size of the buffer, it cannot be used for large data packets, that is, for BACnet/LonTalk.

14.3 KNX architecture (ALN)

KNX is an open standard that conforms to EN 50090 and ISO/IEC 14543. See www.knx.org. KNX corresponds to the former European Installation Bus (EIB) and is backward-compatible.

With KNX technology, advanced multiple disciplines and simple solutions can be implemented to satisfy individual requirements in room and building automation in a flexible way. The ETS, a vendor-independent tool is available for commissioning.

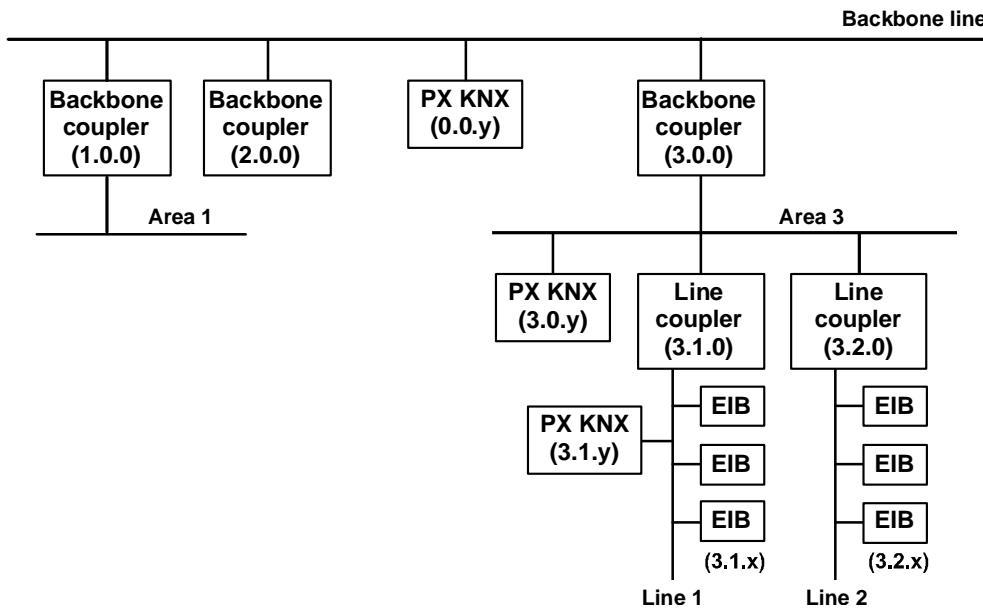
KNX can use twisted pair cables, radio frequency (RF) or data transmission networks in connection with the Internet Protocol for communication between the devices. KNX has links and interfaces for connection to Ethernet/IP, RF, lighting control with DALI and building automation and control systems.

Structure

The following figure shows the structure of the KNX network:

- KNX: KNX devices, e.g., third-party KNX
- PX KNX: Automation station PXC001.D or PXC001-E.D and PX KNX firmware

10523Z16en



Line

A KNX network consists of lines. Up to 64 devices can be connected to each line.

Area

Up to 15 lines can be connected to a main line via line couplers (LC). This is called an area.

Backbone line

The topology can be expanded by means of a backbone line. Up to 15 areas can be connected to the backbone line via backbone couplers (BC). Technically, these are the same devices as line couplers.

Line/Backbone couplers

Couplers separate the areas and lines. Couplers keep the bus traffic within bounds. Datagrams that are only needed on one line should not create a load on the entire network and hence have to be confined to that line. Respective filter tables are created (ETS) when setting up the project/network.

Engineering Tool Software (ETS)

The KNX Engineering Tool Software (ETS) is used to create KNX projects. A bus interface is required to commission the devices with ETS.

For a detailed description of the KNX topology, see

http://www.knx.org/fileadmin/template/documents/downloads_support_menu/KNX_tutor_seminar_page/basic_documentation/Topology_E1212c.pdf.

System Devices

PX KNX

The PX KNX system controller maps KNX devices to BACnet objects. PX KNX also supports different system functions, such as grouping, scheduling, alarming, trending, etc.

The system controller must be positioned correctly in relation to the topology and the load on the bus caused by the devices and connections (group addresses).

Bus power supply

Each line and each area must include a bus power supply.

14.4 KNX PL-Link architecture (FLN)

KNX PL-Link (Peripheral-Link) connects communicating room and field devices (room devices, sensors, actors) with the PXC3 room automation station and the DXR2 compact room automation station. KNX PL-Link fully complies with the KNX standard.

Siemens field devices can be connected to the KNX PL-Link using the KNX PL-Link plug & play capability. KNX PL-Link devices are configured using the Desigo tools. The KNX commissioning software (ETS) is not needed.

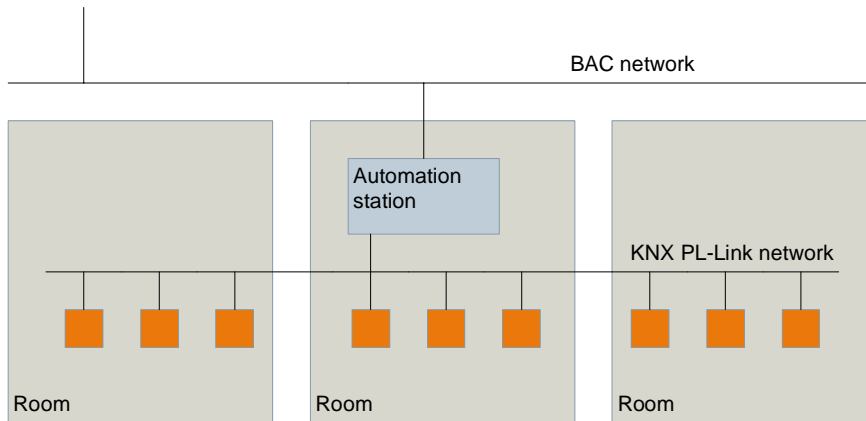
One or more KNX PL-Link devices are connected to the trunk of the corresponding room automation station in a line topology.

A comprehensive library with preconfigured devices supports simple engineering.

The PXC3 room automation station allows for simultaneous integration of devices with KNX PL-Link and KNX S-Mode on a single bus line. Devices with KNX S-Mode are additionally commissioned using ETS.

Structure

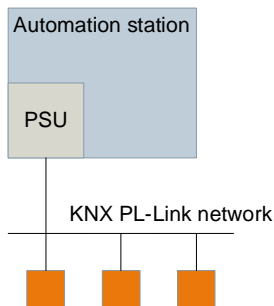
The following figure shows an example of a logical network topology with KNX PL-Link devices, a room automation station and several rooms.



Power supply concept

The PXC3 and DXR2 room automation stations have an integrated KNX power supply to supply their trunks with the corresponding KNX PL-Link devices. This allows simple installations, e.g., an automation station with one or a few room units, without an extra device for power supply to the KNX PL-Link network. If many KNX PL-Link devices are connected, the power supply at the room automation stations is shut off and an external KNX power supply must be used.

The following figure shows the concept of a built-in power supply unit (PSU):



System devices

Third-party KNX devices can be integrated in KNX PL-Link networks via KNX S-mode. The KNX Engineering Tool Software (ETS) is necessary to engineer and commission these devices.

DXR2.M.. automation stations cannot integrate KNX S-Mode devices.

14.5 DALI architecture (FLN)

DALI (Digital Addressable Lighting Interface) is a dedicated protocol for lighting control. See www.dali-ag.org.

DALI is tailor made for modern lighting solutions. A DALI system can be as small as a single luminaire, or can encompass multiple systems across one or more buildings. DALI systems can be connected using lighting hubs/routers.

DALI features:

- Max. 64 devices per subnet (hub/routers)
- Max. 300m cabling
- Max. 250mA device consumption
- Standard two-core cable (1,5mm²)
- Polarity free & free wiring topology

- DALI power and data on the same pair of wires
- Bidirectional communication with feedback of operating state (dim level, lamp failure, etc.)

Structure

A DALI system can be made up of control gear, control devices and bus power supplies.

Control gear

Control gear usually contains the power control circuit to drive lamps, or some other type of output, such as on/off switching or 1 to 10 V analog signals.

Control devices

Control devices can provide information to other control devices, such as light intensity information, and can send commands to control gear. Input devices are a type or a part of a control device that provides some information to the system, such as a button press or movement detection. DALI application controllers are also control devices, e.g., they can send commands to control gear to modify the lighting.

Bus power supplies

At least one bus power supply must be present in a DALI system. This is necessary to allow both communications on the bus, and to power any bus-powered devices. The bus power supply does not need to be a separate unit – it could be part of another device such as a LED driver or a sensor.

Bus wires

A DALI system also includes the bus wires that are used to connect the DALI terminals of the various devices in the system.

Addressing

DALI allows the flexible addressing of devices.

At the simplest level, all devices are addressed simultaneously by broadcast commands. This allows the control of lighting in a similar manner to 1 to 10 V analog control, without requiring any configuration of the individual devices. If a level (Direct Arc Power Command) is broadcast, then all control gear will act upon that command, changing their output to the same new level.

With simple configuration, DALI devices can be given one of 64 short-addresses. This allows individual control, configuration and querying of any single device in the system.

DALI devices can also be group addressed, e.g., a DALI LED driver could be programmed to be in any combination of the 16 available groups. When a command is sent to a group, only devices that are in that group are addressed.

System devices

PXC3...A

The PXC3...A automation stations have a DALI bus for connecting up to 64 DALI ballasts/drivers.

PXC3.E16A

The PXC3.E16A room automation station is optimized for lighting applications. It has an onboard DALI interface for connecting up to 64 electronic ballasts or LED drivers

15 Remote access

The remote access is an access to resources via the internet or a point-to-point connection.

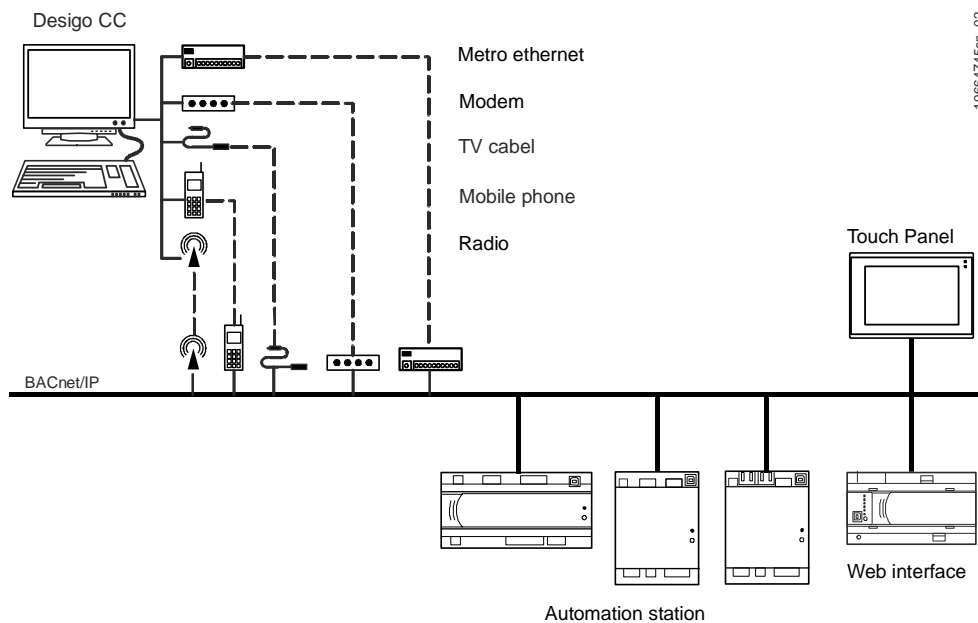
The remote access is used to:

- Connect a remote location to Desigo CC, e.g., for on-call service, managing different locations or support by a specialist
- Remotely access Desigo CC
- Make a change, create an extension or search for errors using an engineering tool
- Forward alarms as text messages or emails from Desigo Control Point or Desigo CC

15.1 Remote access methods

There are two remote access methods:

- Methods that establish a direct point-to-point connection
- Methods that use public networks (e.g., telephone networks for accessing the internet) as a transport medium



The following access networks can be used for the remote access:

- Telephone network
- TV cable network
- Other cable-based networks, such as metro ethernet
- Mobile networks
- Other RF-based access networks

Telephone network-based technologies

DSL variants

Characteristics of DSL variants:

- There are different ADSL and VDSL variants. The DSL variants are country-specific.
- The uplink (that is, the data flows from your private home or project to the internet) and downlink (that is, the opposite direction) bandwidth are different. Take this into account when you select a suitable internet access.
- The DSL line in parallel can be used for telephone calls.
- If you want to use telephony on the same line, you need a splitter in addition to the DSL modem.

TV cable-based access

- This access is similar to DSL. You can access the system remotely via a cable modem provided by the cable network operator.

Other cable-based networks, such as metro ethernet

Characteristics of other cable-based networks, such as metro ethernet:

- Connections with very high bandwidth are available.
- A metro ethernet connection is usually not implemented as part of a BACS project.

Use of mobile telephone networks

The available bandwidth is shared by an unknown number of users with an unknown usage profile. The maximum data transfer rates that are advertised by the mobile network operators deviate substantially from the actual data transfer rates.

The access via a mobile network is less stable than via a cable-based network in terms of availability and data throughput.

If you have to establish a remote access in a remote area, check the service availability and stability. You can use the distance from the base station of the network operator as a criterion. You can also check if there are any large obstacles (mountains, etc.) between the base station and the building.

LTE & UMTS

Characteristics of LTE & UMTS:

- Can be fast

GPRS

Characteristics of GPRS:

- The speed suffices merely for tasks requiring a low bandwidth, e.g., for the system to send an email with a small attachment.

Other RF-based access networks

Characteristics of such RF-based technologies:

- Suited for remote locations, when no DSL is available.
- There are various technologies used by the different providers. Find out what is available at your location.
- Depending on the used frequency, transmission problems can occur during rain or snowfall, even over short distances.

15.2 Choosing a suitable access technology

The technology depends on your intended use and the required bandwidth.

| I want to use the remote access for... | DSL | LTE & UMTS | GPRS | TV cable | Metro ethernet | RF-based |
|---|-----|------------|------|----------|----------------|----------|
| Remote access to Desigo CC | o/+ | o/+ | - | + | + | o/+ |
| Remote access to another BACnet client | + | o | o/- | + | + | + |
| Connecting a Desigo system to Desigo CC | o/+ | -/o | - | + | + | o/+ |
| Alarm forwarding | + | + | + | + | + | + |

Key

- + Good
- o Slow but still possible
- Not possible or too slow

The different access technologies are available with different bandwidth, e.g., DSL (o/+) can be fast or relatively slow.

Costs

The costs are divided into monthly basic costs and usage costs. To optimize costs, analyze your usage profile, that is, how many times per month do you use it and how much data do you exchange per use.

A data flat rate ensures that the costs are capped. Choosing an inappropriate rate plan for a mobile subscription could result in high costs.

Availability

RF-based links and all mobile network-based transmission standards can suffer from transmission problems due to bad weather especially at the cell border. The bandwidth that can effectively be used in the project can vary over the day, because the bandwidth is shared by all users. The bandwidth variations for cable-based technologies are lower.

Recommendations

To ensure a reliable remote access, use cable-based technologies even if the cost is slightly higher. Use mobile networks or RF-based systems only if no alternative is available. If you require a high availability remote access, you can additionally establish a mobile network-based link as a fallback solution. To do this, use a router that offers both a DSL and a GPRS/UMTS/LTE modem.

Every remote access can be attacked. Note the safety measures in the document *IT Security in Desigo Installations* (CM110663).

Access to the PXC..D/-U automation stations via Xworks Plus (XWP) can be protected with a password (password property for remote access [RemAcpwd]). You can enter the password in the Device Property dialog in XWP.

Migrating from an analog modem-based method

Analog modems should not be used in new installations and are not future-proof due to the migration of the networks to Voice over Internet Protocol (VoIP).

ISDN also is not a future-proof technology and should therefore not be used.

If DSL is available, use DSL. Otherwise, use other cable-based internet access networks. If you cannot use such a network, use a mobile network or an RF-based access.

If a project is based on LON, use the PXG3.L router, to connect the remote access on the IP side of the router.

15.3 Technical details

DSL

The DSL modem must match the used xDSL technology and should be purchased in the country of use. DSL connections can use different coding methods, which differ from country to country.

A modem either has one RJ45 connector for connecting the router or has a built-in router. The router must be configured. The modem needs an access code from the Internet Service Provider (ISP).

If the telephone line is to be used for DSL and telephony, a DSL splitter that splits the phone and data signals is necessary.

TV cable-based method

The operator provides the modem. Sometimes, you have to configure the modem. Usually, the cable operator provides a preconfigured modem or the modem configures itself automatically when you connect it for the first time. The modem has an RJ45 connector to connect it to the IP network (the router) or a built-in router. The router must be configured. Sometimes you need to enter an access code received from the operator.

A separate DSL splitter for splitting TV and data signals is not necessary.

Metro Ethernet

Metro ethernet is usually not implemented in a BACS project and is therefore not described in this document.

Use of mobile telephone networks (GPRS/UMTS/LTE)

Several suppliers offer GPRS/UMTS/LTE modems, e.g., modems for private use and modems for industrial applications (also top-hat rail).

Because of the attenuation of the walls and ceilings, the signal inside a building can be weak, that is, an antenna must be placed on the exterior of the building, preferably on the roof.

You can get the best signal strength when the nearest base station of the mobile network you want to use is not too far away and there are no large obstacles between the base station and the modem's antenna (line of sight). Directional antennas improve the transmission quality, but must be optimally directed towards the base station.

The antenna cable between the modem and the antenna must be short, otherwise the signal is too weak. Observe the manufacturer's information on the cable type and maximum length. Antenna cables may not be bent or pinched too severely. The mobile modem must be placed near the optimum antenna location. The length of the cable to the IP network is not that critical.

The mobile network operator provides the SIM card. SIM cards come in various sizes, depending on the modem. Choose the correct SIM card.

The modem is connected to the IP network. The safety measures depend on the modem.

GPRS modems with an RS-232 connection can be connected to some PX controllers using a USB-RS-232 converter.

RF-based access networks

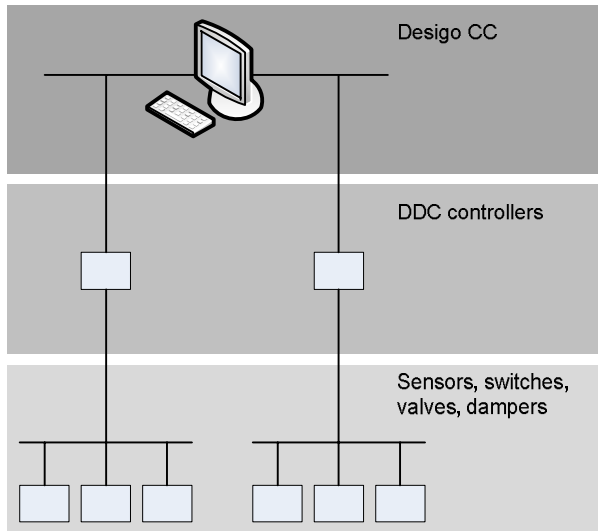
Since there are different technologies, an RF-based access is only implemented in tight cooperation with the network operator. We recommend that you strictly follow his guidelines.

16 Management platform

A building automation and control system encompasses all control functions of one or more buildings.

In addition to typical HVAC systems, there is a need to integrate other areas of the building, such as lighting and blind control systems, fire alarm systems and access systems.

At completion the system comprises one or more superordinate management platforms that let you centrally operate and monitor the individual plants, while each plant's technical building equipment still continues to work autonomously.



Functions

Desigo CC has the following functions:

- *Central operation* of HVAC processes and related areas of a building
- *Visualization, storing and interpretation* of data from underlying levels
- *Control of superordinate functions* (time catalogs, external process reactions)
- *Interface* for external communication (alarm messages, etc.)
- *Data exchange* between DDC controllers (automation level)

Requirements

Modern building automation and control systems need to fulfill the following requirements:

- User friendliness
- Integration capability
- Expandability
- Remote operability
- Cost effectiveness

Advantages

Under the brand name Desigo™, Siemens offers a system family of complementary automation modules and management platforms for buildings and infrastructures of all types and sizes.

Desigo CC has the following advantages:

- A uniform interface for all connected areas from heating, ventilation and air-conditioning through fire alarm systems, video solutions and intrusion alarms to access control systems.
- Cost-effective solutions in every expansion phase through the broad scalability of the number of data points, functions, and broad integration of subsystems.
- A state-of-the-art graphical user interface.
- PC- or server-based management platform based on the current Microsoft operating system.

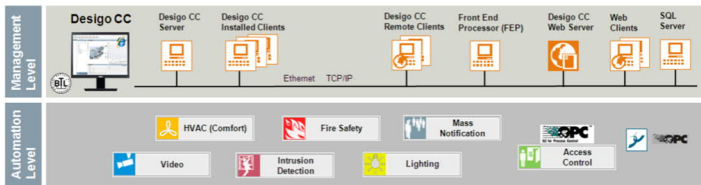
For more information on the Desigo CC management platform, see *Desigo CC System Description* (A6V10415500).

Architecture

The Desigo CC management platform presents a single point of entry for users to operate, monitor and optimize building automation, fire safety and security systems or a combination thereof.

Desigo CC is a flexible, full client-server architecture allowing scalability from small and medium to large and complex systems. The platform provides customizable and market-specific distributions.

Desigo CC can be installed on one single computer, with full server and client functionality. Furthermore, Installed, Web, and Windows App Clients can also be added on separate hardware. Additional system connections can be made through systems installed with Desigo CC Front End Processors (FEP) configurations. Web interfaces provide the customer an increased flexibility for operation and future extensions, e.g. mobile applications for tablets and smart phones.



Main server

The main server contains the project database and the software that monitors and commands the system network. Clients connect to this server to monitor and control the facility. If the same computer runs Microsoft IIS, the installation provides web clients with access to the facility. The Desigo CC server installation always includes an installed client with a user interface for monitoring and controlling the facility. The main server has interface connections to the field (either directly or using FEP) and provides a centralized database and other services to the connected clients. The main server can support a number of clients that are connected using a network (LAN) or Intranet (WAN).

Installed client

The Installed Client is typically used for operators who are focused entirely on monitoring and managing building systems. In this configuration, software components used for event management are locked in place and cannot be moved or covered by other applications. This ensures that critical events are never missed or hidden. Installed Clients can optionally be configured to run in a closed mode where only Desigo CC and other specifically identified applications are allowed to run. In closed mode, the workstation is dedicated to running Desigo CC, with access to the Start menu or other operating system and customer applications available only to administrative users.

Web client (browser client)

The web client is deployed on the intranet with full trust and allows access to local resources. The system runs in the Internet Explorer browser (using HTTP or HTTPS as communication protocol) and is downloaded on demand each time the user launches the system as web application. When working in a browser, you can have the same capabilities as those working on an Installed Client, or can be restricted to have different access when connected remotely.

As web clients require low latency and high network bandwidth, they are appropriate for intranet use. We do not recommend it for internet use.

See *Desigo CC Installing the Web Client Application Certificate (A6V10415479)*.

Windows app client (ClickOnce)

The Desigo CC Windows App Client looks like the standard system software, but is a light application that can be downloaded from the Desigo CC server when connecting through a browser. When the Windows App Client is downloaded, it runs like any other Windows application on the desktop. It can be launched from the Start menu, desktop icon, quick-launch toolbar, and so on. This deployment does not require administrative privileges. The Windows App Client runs in its own pane, without the overhead of the internet browser application and menus.

Web server

To use the Desigo CC Web and Windows App Clients, you must install the web server. To install the web server, you must first install Microsoft IIS on the web server computer. Usually the web server is on the Desigo CC server. It might be located on a separate computer, if the customer's IT department requires the web server to be installed in a separate controlled environment, or if it is preferred not to use the resources of the system server for the Microsoft IIS tasks.

The web server lets you to access the system using the intranet and a web browser. You can add only one web server. It lets you download all files required for the Web Client and Windows App Client environments. It provides a system web page to access the Web Client, the Windows App Client, and the system documentation in the Internet Explorer browser. It also represents the endpoint of the communication with the system server.

Front End Processor

A Front End Processor (FEP) is a computer that provides additional connections between building level devices (such as field panels) and Desigo CC. By providing additional connections to the building level network, an FEP enables load balancing for the network-based processing for a Desigo CC system.

System Dimensioning Guide

Desigo CC covers a wide variety of solutions so that it is impossible to define simple rules for determining the size. Therefore, a system dimensioning tool is available that estimates the system size and disk storage space on the basis of information available at the time of the offer, e.g., the number and type of physical points and the expected history data base contents.

16.1 User functions

Graphics

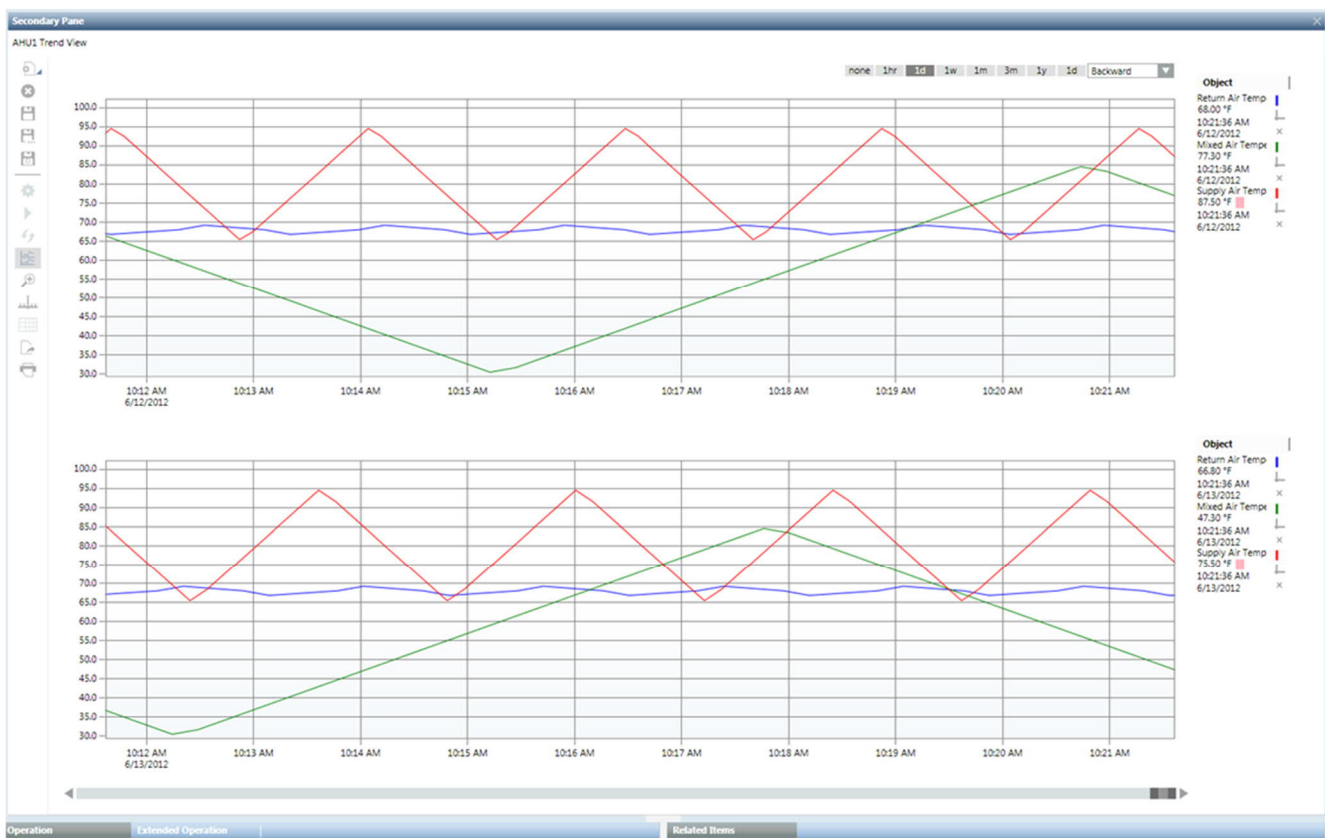
The Graphics application allows you to view the configured graphics representing your facility or equipment. You can change the current state of an object's properties from a graphic, filter your view of a graphic by discipline and section and you can zoom in and out for greater detail or for a birds-eye overview.

Trends

All available process data of a system can be recorded and applied to operational optimization. This lets you record information on plant states, temperature curves, switching states, and counter values in a form that is suitable for your purposes. The measured value data can be displayed and evaluated graphically.

Online trend records real-time values from your plant and displays them graphically in a Trend View. If a value changes, the data values are sent to the trend application. Offline trend data is used for the longer-term storage and retrieval of historical data for the analysis of entire plants or single processes. With offline trend, data is recorded directly in the automation station.

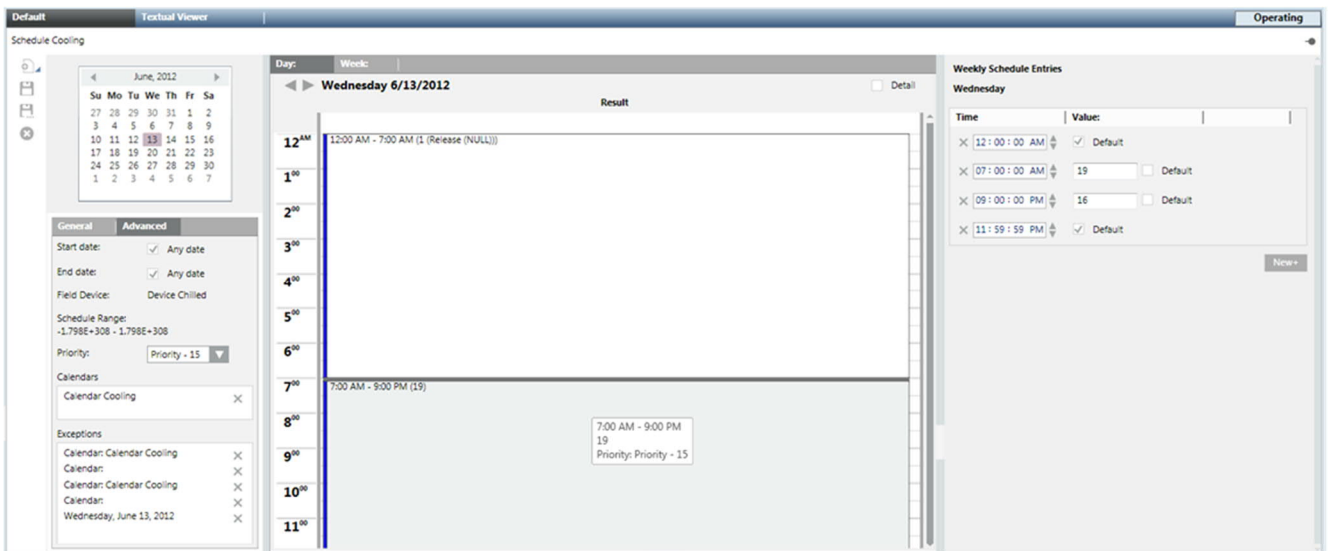
Trend and system activity data is stored in a Microsoft SQL Server database. Microsoft SQL Server Express is included with the Desigo CC software, and can be upgraded as required. The Trend Comparison View lets you time-shift the trend view to compare data at different times for quick analysis of changing conditions.



Scheduling

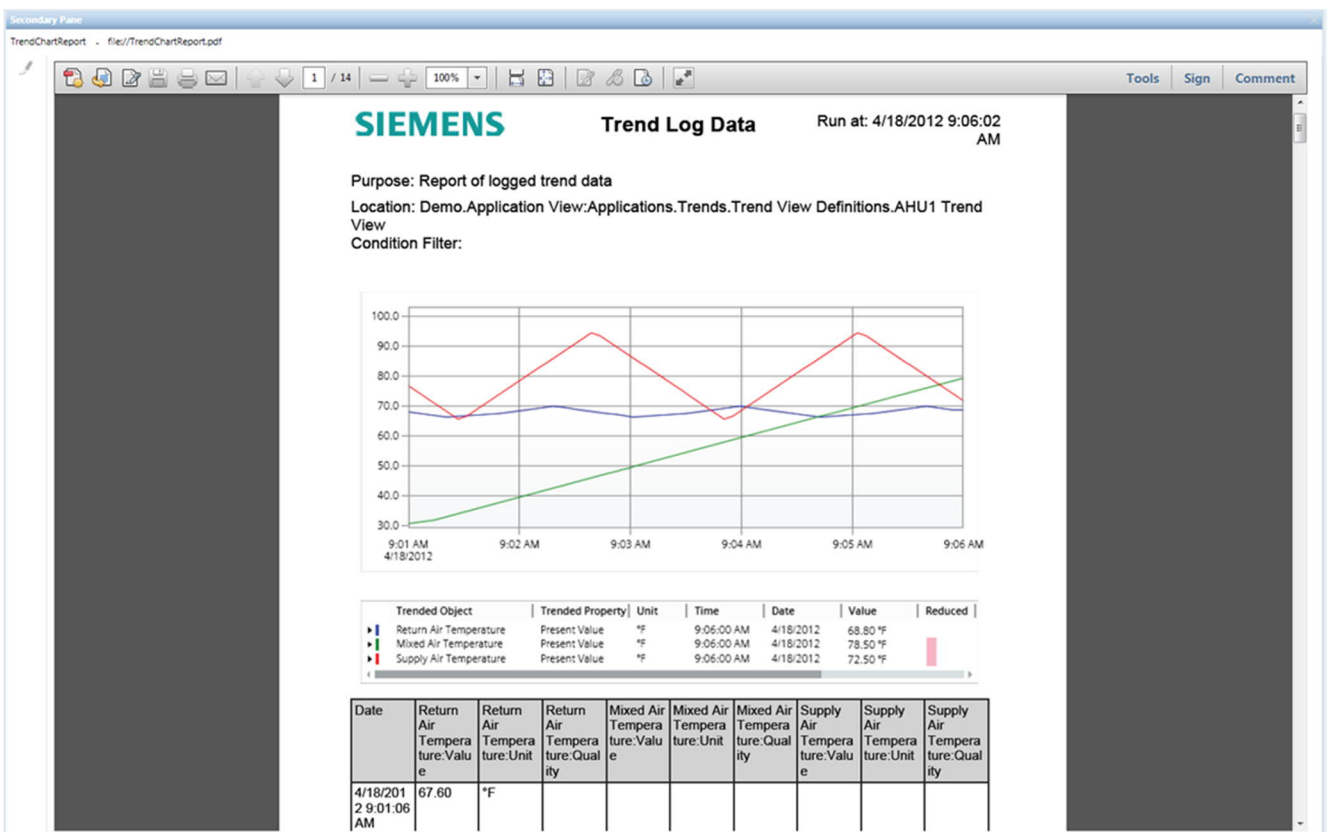
The Scheduler allows you to schedule events for Desigo CC and field panels at your facility. You can create daily or weekly schedules for Desigo CC and BACnet devices. You can fully configure and monitor standard BACnet schedules, calendars, command objects, and workstation-based schedules that can be used to support systems without built-in scheduling capabilities. Schedules are automatically associated with systems they

control, so you can quickly navigate to the schedules of any selected object. A Timeline Viewer lets you view the details of multiple Designo CCs and field panel schedules simultaneously, spanning a range of time.



Reports

The Designo CC reporting tool includes standard reporting templates and lets you create fully configurable reports with custom logos, headers, footers, and layouts that include tabular and graphical system information. You can schedule reports and save them in CSV or PDF formats.



Event management

Event management allows you to manage events throughout the system. You can monitor and manage the progress of each event from initiation through resolution. The full history of each event issue is recorded, and you can generate event-related reports that you can view, save, and print.

Log Viewer

The Log Viewer application provides an historic log of all user and system events and activities that have occurred. You can retrieve these historic events and activities for further analysis and investigation using sorting and filtering. Log views can be saved and exported if required.

Detailed log

The detailed log allows users to view the most recent records for any selected object. The same content filtering and sorting functionality available in the Log Viewer is possible in the detailed log.

Remote notification

You can configure Desigo CC to automatically or manually send email or SMS messages to specific recipients. You can specify:

- What events the recipients should be notified for and when
- How notifications are escalated from one recipient to another until a notification message is responded to
- If a message is periodically sent to the operators stating that the system is running normally
- Which devices are used for the notification

Macros

Macros are predefined lists of commands that enable a user to send out a group of commands to specified devices with a single action. Some macros can be started manually while others may be part of schedules defined for time-based functions or automatic reactions. Macros are also used by the system to perform multiple command actions. These predefined system macros are applied to specific control actions, such as block commands to fire control panels and system backup functions.

Reaction Processor

Reactions are automations programmed in the system, so that when a specific situation occurs on site, a command or a series of commands is automatically executed.

You can define actions to be executed automatically when specific conditions are verified. Conditions can be based on time, on events, on a change of values, or on a combination of some or all. When conditions are met, the Reaction Processor executes a pre-configured list of commands.

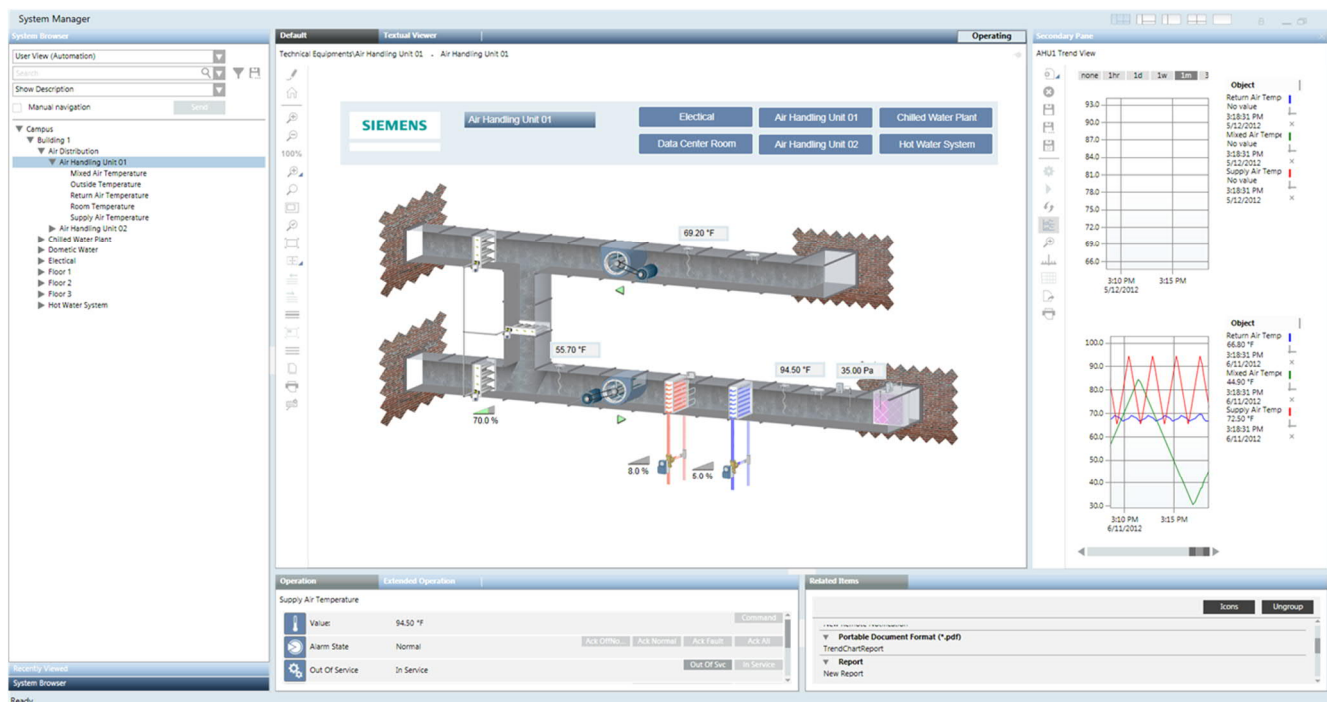
Document management

Desigo CC can handle the different types of document templates used in the project. You can configure document templates in PDF, RTF, TXT, XLS, and HTML format.

16.2 Main components

System Manager

The System Manager lets you navigate the system, view and override current conditions, analyze historical operations, and configure the system. The System Manager contains the System Browser, Primary, Operation and Related Items panes that interact via built-in workflows. Multiple system management session can be concurrently used.



System Browser

The System Browser displays objects in the building control system through various views. You can search and filter objects, display object names and descriptions, and drag objects into Trends, Schedules, and Reports.

History Database (HDB)

Historical data is stored in an access-controlled MS SQL Server database. The System Management Console lets you create a project History Database (HDB) and link it to the active Desigo CC project on the main server. The history database is used to log a wide range of user and system activities, such as:

- User and system activities
- Alarms and their treatment
- Faults that have occurred and are handled as batch messaging
- Values that are logged as trends

Project database

The runtime data (process image) and the engineering data are stored in a file-based database in a subdirectory of the project directory. The data is unencrypted and database access can only be prevented by restricting access to the database files. The project directory needs to get shared when deploying installed clients. It is therefore important to restrict access on the *db* folder in the project directory to the Windows account running the Desigo CC main server.

Microsoft SQL Server

Desigo CC uses the Microsoft SQL database software. Microsoft SQL Express is included on the product installation DVD (Microsoft SQL Server 2008 R2 Service Pack 2, Express Edition, version 10.50.4000.0). Alternately, you can use an existing Microsoft SQL Server installation (same version 10.50.4000.0). In this case, the Desigo CC Installer will skip the Microsoft SQL Server installation. In both cases, Microsoft SQL must first be installed and running on the computer where the Desigo CC main server will be installed.

Microsoft IIS server

A Microsoft Internet Information Services (IIS) server for Web Clients and Windows App Clients can be installed on the Desigo CC server or on a separate installation (web server).

License Manager

Licensing ensures the operation of the system within the agreed system limits. Only the system is allowed to change license data.

If a license becomes temporarily unavailable (e.g., due to network connection issues) the system continues to run fully operational for a grace period. The system continues to check for the license and shuts down at the end of the grace period, if none of the license checks succeed.

Exceeding the limits of the license (e.g., by integrating more field system data points than stated in the license) puts the system into courtesy mode. Phases of courtesy mode accumulate until a total duration of 30 days is exceeded, then the server shuts down. Unless new licenses are made available, after a manual restart the system again goes into courtesy-mode exceeding and shut down.

Any unauthorized attempt to modify system license data directly in the database (e.g., changing the remaining time of a specific license mode) shuts down the system.

16.3 Access and security

User management

User privileges can be assigned to users and to workstations, allowing users to be granted the same access from everywhere or different access depending where they're logged on. The user interface displays only elements, such as menus, buttons, list items, tree nodes, where the user has at least read access.

Access privileges can be assigned to resources/groups, such as workstations, features, applications, system objects, system object properties and logical groups of these resources.

User authorization

User access rights in Desigo CC are determined by four main factors:

- The system must know the user (authentication).
- The user must be assigned to a user group.
- The user must have the appropriate application rights.
- The user must have the appropriate scope rights.

If all of these conditions are met, the user can log on to Desigo CC, and read/write objects and execute tasks, depending on the assigned rights.

See *Desigo CC Engineering Manual* (A6V10415473).

Scopes

Scope is the general term for specific object access in Desigo CC. A scope segments and implements certain rules for the user role in the project. A user only sees the area of the building assigned to him, e.g., pumps, receives only alarms from this area in the event of an emergency and can only acknowledge those alarms. If an emergency occurs in an area that is not in the scope of this user, e.g., ventilators, the user does not receive an alarm about this event.

Communication security

In general, communication channels are non-encrypted due to performance reasons. Exceptions are communication channels for file transfer using web and video transfer. Sensitive data (passwords during authentication or user management configuration) is transferred as encrypted message content.

Wireless input devices (especially keyboards) use radio transmission that is often not or inadequately cryptographically protected. Even from greater distances, it is possible to listen in or even plant external data in the system.

We recommend that you do not use wireless input devices. If you must use wireless input devices, use only devices with proven encryption.

Communication ports and protocols

Which ports are used depends on the actual deployment and subsystem integration of the whole system.

See *Desigo CC System Description (A6V10415500)*.

16.4 Event management

Desigo CC lets you quickly, easily, and accurately respond to any event.

Summary Bar

The Summary Bar contains a summary of the events occurring in the system and lets you quickly access functions, such as the Event List. It also displays information, such as the system status, the logged in user, etc. Depending on the client profile in use, the Summary Bar can be docked on the desktop or freely opened and closed as needed.

Event List

The Event List provides a complete and easily filtered list of events under control of Desigo CC. When the Event List is expanded, it clearly shows each event source, severity, current status, custom messages and suggested action steps through the use of text, color, and icon representations. You can acknowledge, silence, and reset alarms from the Event List.

| Cause | Source Status | Source | Commands | Event Status | Date/Time | Counter | Category | Tag |
|--------------------|---------------|----------------------------------|----------|--------------------|-----------------------|---------|-------------|-----|
| Off Normal (Quiet) | Active | Life Safety Zone 1 | ✓ | Unprocessed | 5/7/2012 6:26:43 PM | | Life Safety | |
| Off Normal (Quiet) | Quiet | Life Safety Zone 2 | ✓ | Unprocessed | 4/5/2012 12:14:31 AM | | Life Safety | |
| Off Normal (Quiet) | Quiet | Life Safety Zone 4 | ✓ | Unprocessed | 3/29/2012 1:26:17 AM | | Life Safety | |
| Off Normal | Quiet | Solar Collector Temperature | ✓ | Unprocessed | 3/27/2012 10:09:33 PM | | Alarm | |
| Off Normal | Quiet | Cooling Tower Return Temperature | ✓ | Unprocessed | 3/27/2012 10:07:29 PM | | Alarm | |
| Off Normal | Quiet | Supply Air Temperature | ✓ | Unprocessed | 3/27/2012 10:07:22 PM | | Alarm | |
| Off Normal | Quiet | Hot Water Supply | ✓ | Unprocessed | 3/27/2012 10:09:03 PM | | Alarm | |
| Off Normal | Quiet | Mixed Air Temperature | ✓ | Unprocessed | 3/27/2012 10:04:43 PM | | Alarm | |
| Off Normal | Quiet | Boiler Water Supply Temperature | ✓ | Unprocessed | 3/27/2012 8:58:55 PM | | Alarm | |
| Off Normal | Quiet | Solar power | ✓ | Unprocessed | 3/27/2012 8:58:55 PM | | Alarm | |
| Off Normal | Quiet | Voltage Phase A to Neutral | ✓ | Unprocessed | 5/7/2012 6:37:16 PM | | General | |
| Off Normal | Quiet | Current Phase A | ✓ | Unprocessed | 3/28/2012 6:01:28 PM | | General | |
| Off Normal | Quiet | Voltage Phase B to Neutral | ✓ | Unprocessed | 3/27/2012 10:11:40 PM | | General | |
| Fault (Quiet) | Quiet | Life Safety Zone 1 | ✓ | Unprocessed | 5/7/2012 6:28:29 PM | | Fault | |
| Fault (Quiet) | Quiet | Life Safety Zone 6 | ✓ | Unprocessed | 4/4/2012 12:44:49 AM | | Fault | |
| Off Normal (Quiet) | Quiet | Life Safety Zone 6 | ⊖ | Ready to be closed | 4/4/2012 12:45:57 AM | | Life Safety | |
| BACnet Driver Info | Quiet | Device AHU 02 | ⊖ | Ready to be closed | 6/14/2012 3:36:34 PM | 9 | Status | |
| BACnet Driver Info | Quiet | Device Room | ⊖ | Ready to be closed | 6/14/2012 3:36:34 PM | 9 | Status | |
| BACnet Driver Info | Quiet | TEC Application 603 | ⊖ | Ready to be closed | 6/14/2012 3:36:34 PM | 9 | Status | |
| BACnet Driver Info | Quiet | TEC Application 2027 | ⊖ | Ready to be closed | 6/14/2012 3:36:34 PM | 9 | Status | |

Event Bar

When using profiles for critical event management, you can collapse the Event List into a condensed list of event buttons in an area called the Event Bar, that remains docked on the desktop for easy access. This lets you keep an eye on the current situation at all times.

Client profiles

To ensure the correct level of event management support for users in any situation, a workstation and/or users can be easily assigned predefined profiles supporting casual, intermediate, or dedicated event notification and management.

Fast treatment

From the Event List or Event Bar, you can quickly select an event and perform all the commands (e.g., Acknowledge, Reset, Close or Suspend) from the Event Detail Bar and Event List, without looking at treatment steps, viewing live video or a map of the alarmed area, etc. The event descriptor (visible when the Event List is expanded) contains a short description of the next action (which command to select).

When event treatment is in progress, you can send the available commands to the source object causing the event or suspend treatment.

Investigative treatment

From the Event List or Event Bar, operators can quickly open the System Manager with focus on the source of the event, and all information (live video, recent history, schedules, etc.) related to the event source.

Operating procedures

Operating procedures consist of a sequence of steps or actions, which the operator must, or is suggested to perform with the assisted treatment. For each step of a procedure, the system provides instructions and operating tools. With appropriate permissions, you can create, view, edit, or delete operating procedures.

Assisted treatment

From the Event List or Event Bar, operators can quickly open the assisted treatment to guide the operator through pre-configured operating procedures. Each operating procedure is composed of steps - some of which may be mandatory - for the user to complete (e.g., to see the graphic of the object in alarm, fill-in a treatment form, or automatically print the information of the event).

16.5 Installation, setup and engineering

License Management Utility (LMU)

The installation program installs the Siemens License Management Utility (LMU) on every management platform in a Desigo CC network. The LMU enables and manages licenses and holds the installed licenses for Desigo CC. The operating state of Desigo CC, the number of seats, the point count, and all functions are controlled through the LMU. Each Desigo CC management platform must be licensed locally. Licenses can be activated, repaired, returned and renewed through the LMU.

After you install the LMU, you must activate the Desigo CC licenses using the following licensing methods:

- Online: Licensing carried out via the internet or intranet on the back office license server.
- Certificate/Dongle (including remote dongle engineering): Licensing carried out via certificate files representing the license.
 - For Dongle-bound licenses, dongles and licenses can be obtained individually and subsequently tied to each other and loaded onto the PC.
 - Engineering licenses are always dongle-bound. Where a physical connection of the dongle to the PC is not possible, e.g., during a remote support session, the engineering license can still be used for a limited time.
- Manual: Manually returning a license based on XML request/response files.

See *Desigo CC Installation Manual* (A6V10376166).

System Management Console (SMC)

The Desigo CC server hosts the System Management Console (SMC), a stand-alone tool which is installed on the main server only, and can only be launched locally. Once Desigo CC is installed successfully, the field engineers must first perform the typical system administration operations, such as configuring system users, projects, and history database in the SMC, before being able to launch a Desigo CC client.

Profiles, schemas and templates

Client profiles define the appearance and behavior of the system functions involved in event management, such as Summary Bar, Event List, Event Detail Bar, event filters, and event treatment. Every project template has a matching client profile, and every client profile has a matching event schema. To ensure a consistent configuration, the project template, the client profile and the event schema must match.

Subsystem integration

Representative data points in Desigo CC can be created manually, imported through data exchange files, or uploaded through a selective auto-discovery mechanism depending on the type of system being connected. A unique, extensible object modeling approach allows Desigo CC to normalize information brought in through any interface, and to provide the same look, feel, and operation through a common set of applications, without concern for the source of the data.

Desigo CC lets you configure connected subsystems directly and perform typical automation station functions, such as scheduling and event generation, at the management platform for connected systems that do not support those functions directly.

Desigo CC supports the following subsystems:

- Desigo building automation system
- Desigo room automation system
- Siclimat-X
- Sinteso Fire Safety System
- Sinteso Fire Safety System
- Intrunet Intrusion System
- Video through Milestone Video Management System
- Mass Notification System (see MNS documentation)
- Third-party Building Automation and Fire Safety systems based on BACnet/IP
- Third-party subsystems through OPC
- Third-party subsystems through Modbus/IP
- Integration through SNMP
- APOGEE Building Automation system
- XNET FireFinder XLS and MXL fire safety systems
- Desigo Fire Safety FS20 UL systems

Auto discovery

Auto discovery lets you discover and import devices, which are already on the network, into Desigo CC. You can set filters and detect your devices on the network, which then display in the System Browser. You would typically use this method for existing jobs, where field panels are already installed and online.

OPC server

OLE for Process Control (OPC) is a widely accepted industrial communication standard that enables the exchange of data between multi-vendor devices and control applications without any proprietary restrictions. OPC is a client-server technology and Desigo CC can act as the server providing data to third party clients.

Web services

Using RESTful technology, Desigo CC provides alarm, object and time series data via web based services to supervision management platforms or other third-party external applications.

Language packs

The Desigo CC software is delivered in English and can be extended with additional languages. The following software language packs are supported:

- Arabic
- Chinese (simplified)
- Chinese (traditional)
- Czech
- Danish
- Dutch
- English (default)
- Finnish
- French
- German
- Italian
- Korean
- Norwegian

- Polish
- Portuguese
- Russian
- Spanish
- Swedish
- Turkish

You can install 3 languages simultaneously. Every user can define his user interface language.

Project and HDB backup

Backing up Desigo CC requires saving independent parts on different servers or PCs. We recommend that you save the backups of your project data to a different machine from where they originally reside.

Two parts must be backed up:

- The entire customer project data, including all libraries, configurations, object data (project backup).
- The historic data collected in the history databases (HDB backup).

Backups can be done either manually or by applying a macro in combination with a management platform scheduler.

See *Desigo CC System Management Console* (A6V10415497).

16.6 Graphics libraries

Desigo CC contains libraries with symbols and graphic templates for easy plant graphics engineering. The Graphic Library Browser shows all the available symbols and graphic template objects in your project libraries.

Symbols

A graphics symbol is a reusable graphic image that represents a piece of equipment, floor, or any component or entity. Symbols are stored in a library and are used to display system object values. Symbols can be associated with one or more object types in the Models & Functions application and bound to object type properties to create substitutions in your graphics that provide a dynamic, visual representation of changing values from Desigo CC.

In its simplest form, a symbol is a graphic made up of drawing elements on the graphic canvas in the Graphics Editor. Each drawing element has a series of associated properties. These properties can be used to create substitutions. Symbols can be associated with an object type

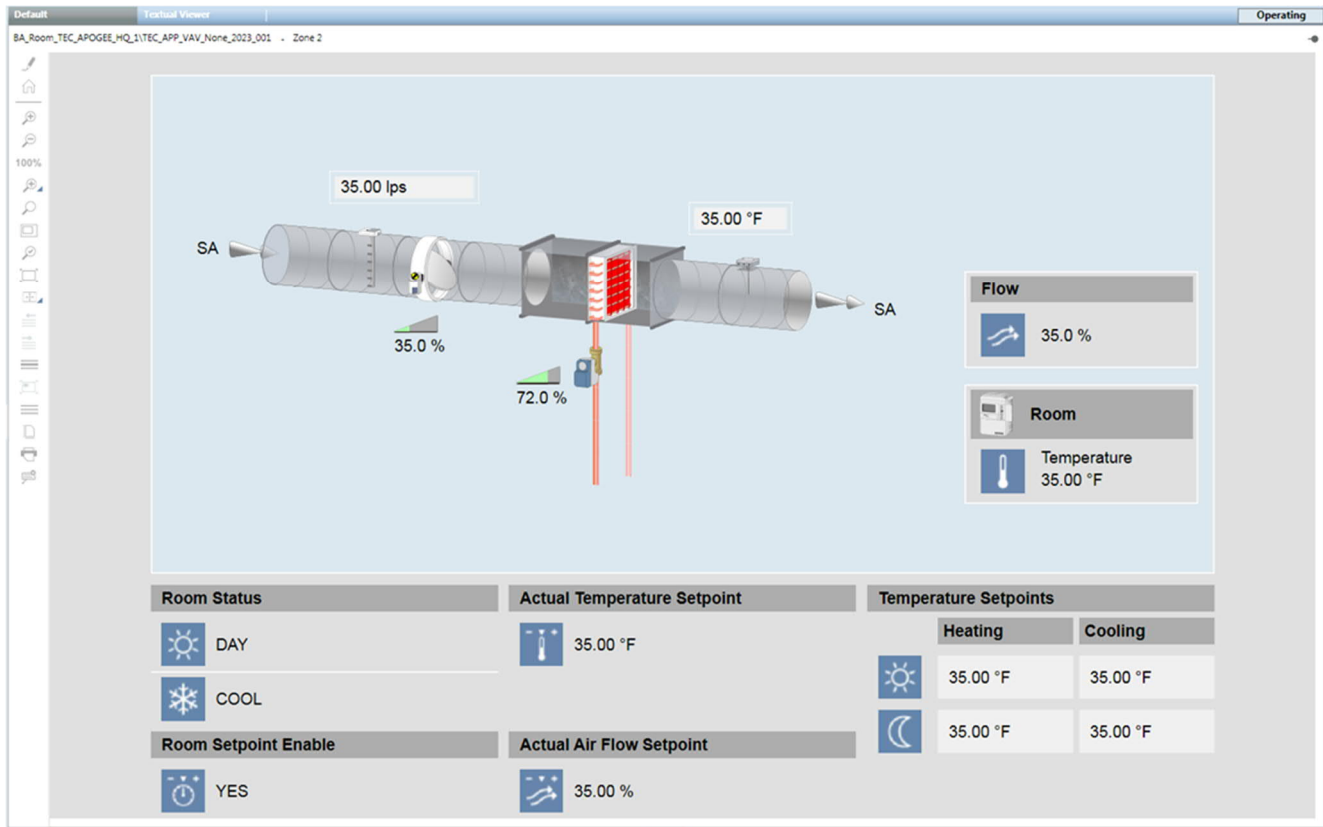
An object type is associated with a symbol in the Models & Functions application. When you drag-and-drop the symbol onto a graphic, the symbol displays the system object values in runtime mode and in the Graphics Viewer. Animation is supported through a series of graphics. Pre-defined symbols are stored in library folders. These symbols are visible and editable from the Graphics Library Browser. Advanced users can create their own symbols.

Generic symbols

A generic symbol is a concept that allows you to create one type of symbol that can support an object that has one or several properties with changing values. Depending on the object, the symbol will not display the elements of the graphic that do not have a data point associated them.

Graphic templates

Desigo CC provides standard BACnet TEC graphics for various applications. You can also create template graphics for TEC applications.



See *Desigo CC Getting Started* (A6V10415475) and *Desigo CC User Guide* (A6V10415471).

16.7 Graphics engineering

Desigo CC graphics are built using smart objects that know how they are used and how to represent themselves graphically. Smart objects let you create graphics by dragging objects onto a page, without manually binding an object to graphical symbols.

The Graphics application allows you to create, view, store, and command large graphics representing equipment, floors, buildings, facilities, and entire campuses. These graphical representations can contain dynamic elements to represent devices or values you want to monitor or control. The Graphics application consists of:

- Graphics Viewer
- Graphics Editor
- Graphics Library Browser

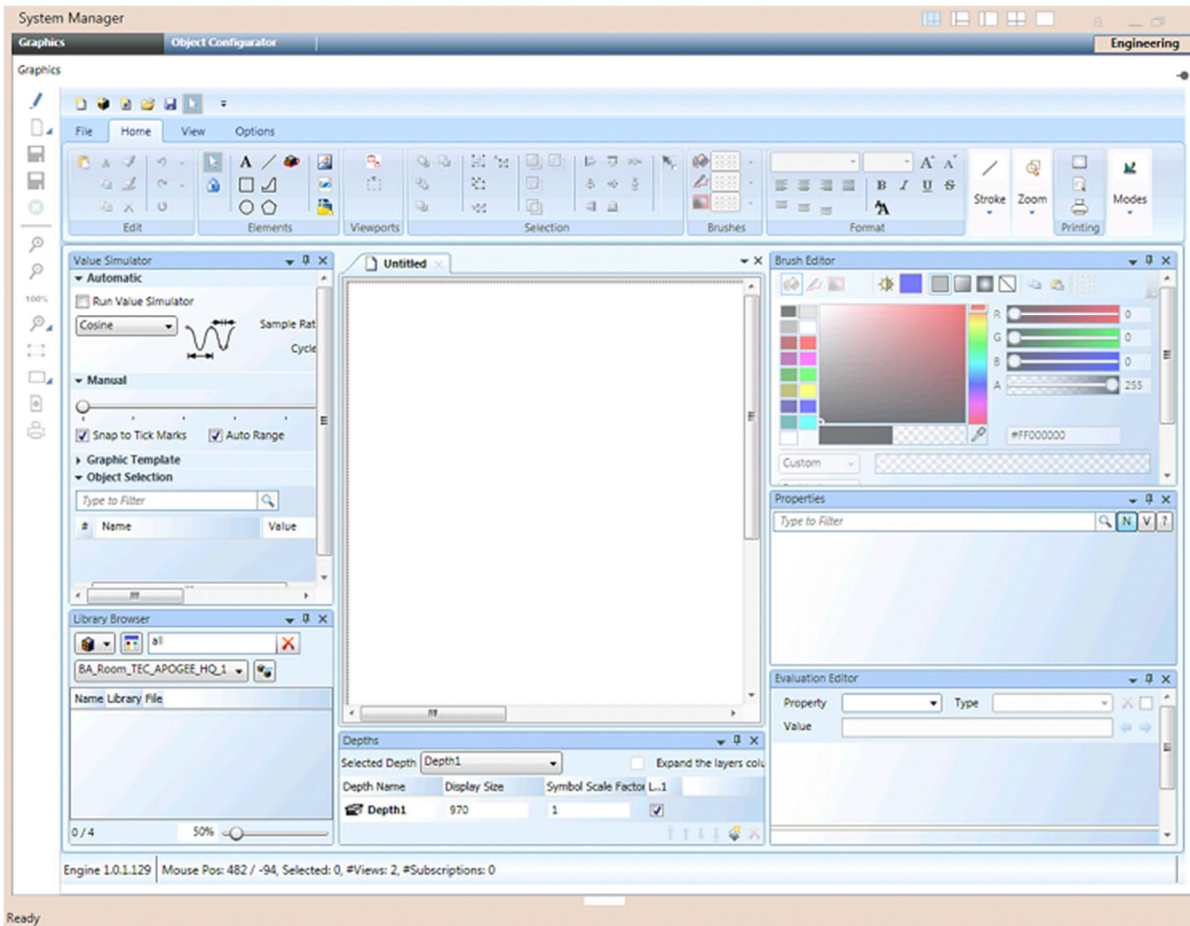
Graphics Viewer

The Graphics Viewer lets you view the graphics representing your facility or equipment. You can change the current state of an object's properties from a graphic. You can filter your view of a graphic by discipline, section, or you can zoom in and out for greater detail or for a birds-eye overview.

Graphics Editor

The Graphics Editor lets you create dynamic graphical representations of your plants, buildings or equipment. You can test and simulate your dynamic graphics before going online with them.

See *Desigo CC Graphics Editor* (A6V10415487).

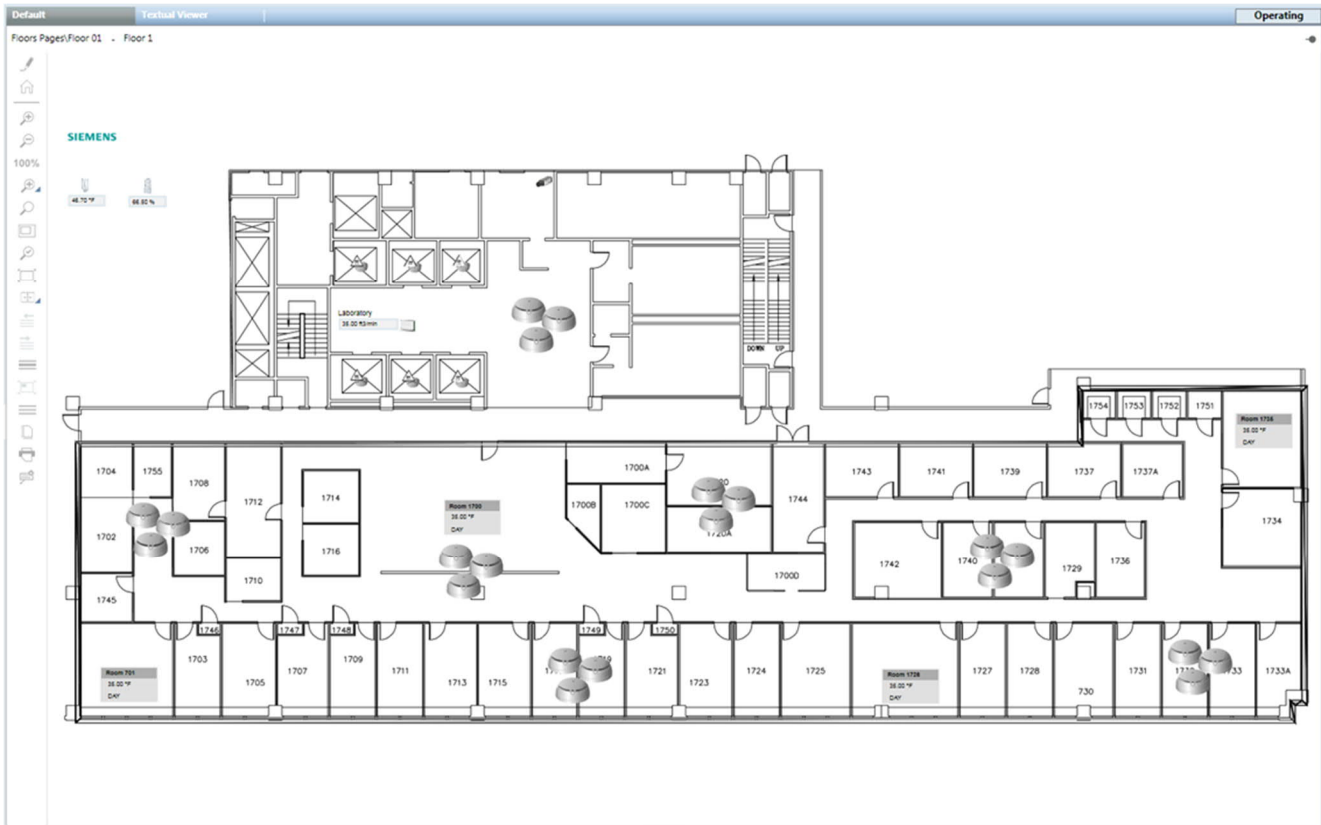


Graphics Library Browser

The Graphics Library Browser lets you toggle between a view that displays all the available symbols and graphic template objects in your project libraries.

AutoCAD import

You can import AutoCAD drawings and select and manipulate the layers of the AutoCAD drawings during and after the import process.



16.8 Virtual environment

Desigo CC is compatible with following Virtualization software packages:

- VMware®:
 - Virtualization platform: VSphere 6.0
 - Fault-tolerant software: ESXi 6.0b (build 2809209) managed by VCenter Server Appliance v6.0.0 (build 2793784)
- Stratus®:
 - Virtualization platform: KVM for Linux CentOS v7.0
 - Fault-tolerant software: everRun Enterprise 7.2
 - Virtualization platform: Citrix XenServer 6.0.2
 - Fault-tolerant software: everRun MX 6.2 HotFix4 (build 6.2.9125.825-HF:EA)

17 Desigo Control Point

Desigo Control Point is an embedded building management station for operating and monitoring building automation and control systems on BACnet/IP.

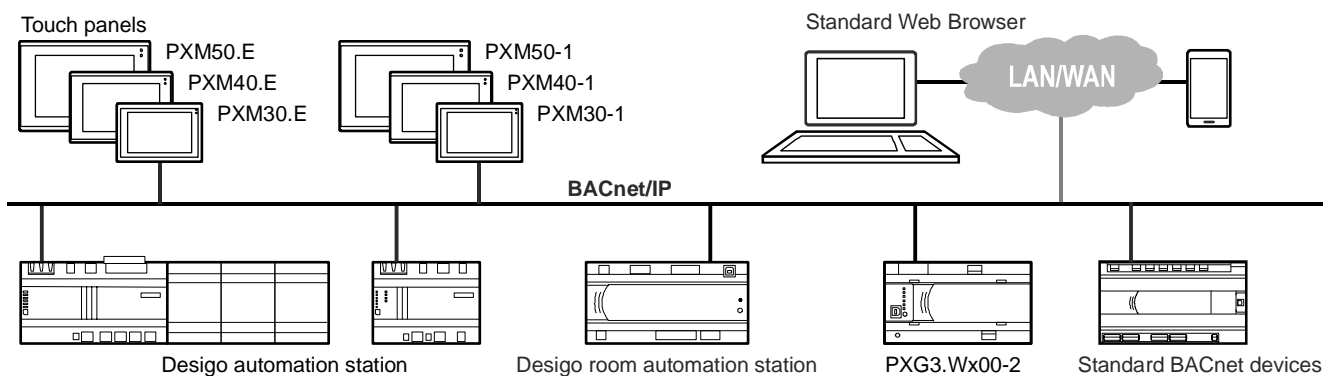
Additionally room applications can be operated by the end user (using QMX7 widgets).

The functionality can be adapted to any user profiles – from room users to facility managers.

Operable systems

- Desigo primary plants
- Desigo room automation
- BACnet third-party devices and systems

Topology



PXM30-1, PXM40-1 and PXM50-1 touch panels

The touch panel clients are used on projects that require multiple touch panels or third-party devices to operate the same system data. The central web interface PXG3.Wx00-1 can connect multiple devices.

Use:

- Multiple touch panels on a plant room, conference room, open office spaces, etc.
- A touch panel in the plant room for on-site operation, remote access via third-party devices.

PXM30-E, PXM40-E and PXM50-E touch panels

The BACnet/IP touch panels can be connected directly to the BACnet network. No additional devices are required to operate the Desigo system.

In addition, the BACnet/IP touch panel also includes a web interface for remote operation using a standard web browser on third-party devices. The functionality of the web interface essentially matches the functionality of the PXG3.W100-1, differing only in the permissible system limits. The integrated web access increases the competitiveness of Desigo on smaller projects since there are no additional costs and installation effort.

Use:

- Very small projects: One touch panel for on-site operation of a primary plant. Remote access occurs directly via the integrated web interface on the touch panel.
- Large project with multiple, decentralized touch panels. The central, cross-building operation takes place on Desigo CC.

PXG3.W100-1 and PXG3.W200-1 BACnet/IP web interfaces

The BACnet/IP web interface permits local and remote operation of Desigo primary and room automation stations as well as third-party BACnet/IP devices. The products PXG3.W100-1 and PXG3.W200-1 vary in functionality as well as permissible system limits.

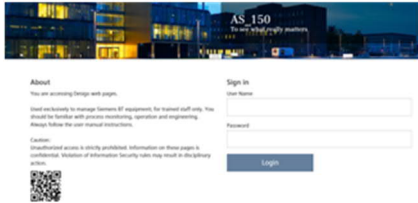
Use:

- Remote access to the plant.
- Local operation using third-party devices.

17.1 Functions

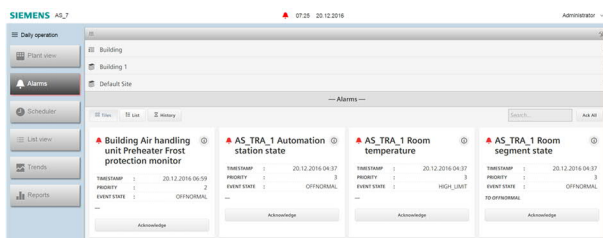
User management

- User logon function to protect access.
- Automatic logoff function (can be disabled as an option).
- Online user management: Create, edit, and delete user accounts.
- Data point access based on user rights.



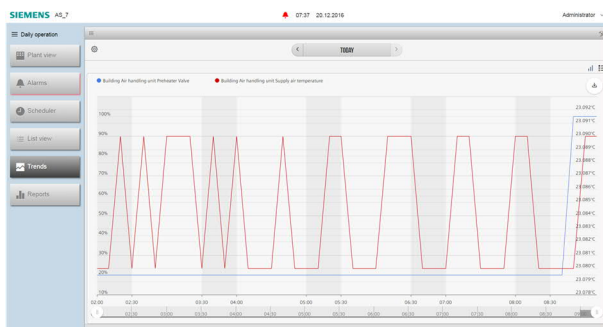
Alarming

- Event list for the generic display of alarms and events, regardless of whether they are included in a user view.
- Trend and display historical, confirmed alarms and events.
- Forward alarm messages to multiple e-mail or SMS recipients. An SMS gateway on the Internet is used for alarming via SMS.
- Alarms are forwarded pursuant to a configurable schedule and based on alarm priority.
- The email subject line with alarm message can be configured.



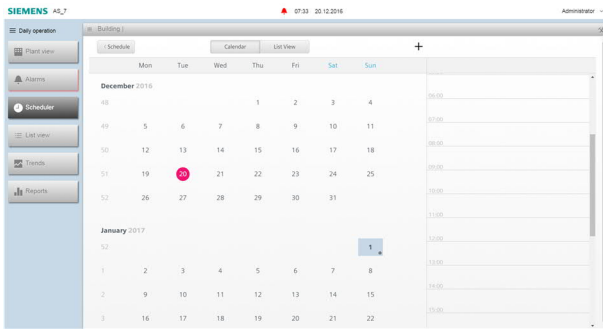
Trend

- Graphic display of multiple trend objects on a trend graph on two Y-axes.
- Trend objects can be read from remote Desigo PX automation stations (offline trend) or created during runtime on the device, dynamically (online trend).
- Highly flexible configuration and modification to trend view.
- Manual export of trend data via CSV file.
- Automatic transmission of trend data via email or save to an FTP server, pursuant to a schedule.



Schedulers

- Operate BACnet schedulers and calendar programs.
- Supported scheduler program objects: Analog, binary, and multistate.
- Operate local and global calendars on Desigo PX automation stations.
- Create and copy efficient exception programs easily.



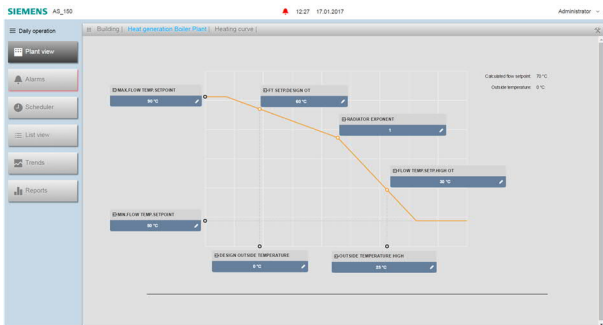
Reports

- Filter data points by alarm state, operating state, or object type.
- Manually export reports via CSV file.
- Transmission of reports via email or saving to an FTP server.

| DEVICE ID | SOURCE | NAME | VALUE/STATUS | STATUS | PRIORITY |
|-----------|---|----------------------------|---------------------|--------|----------|
| AS_7-001 | Building Air Handling Unit Temperature con... | Outdoor High Supply air | 32 °C | normal | |
| AS_7-002 | Building Heat generation Boiler Plant Heat... | Outdoor temperature | | normal | |
| AS_7-003 | Building Heat generation Boiler Plant Heat... | Base point for DP | 33.000000/170728 °C | normal | |
| AS_7-004 | Building Heat generation Boiler Plant Heat... | FT output at transition | 32 °C | normal | |
| AS_7-005 | Building Air handling unit | Heating to the alarm limit | 1 | normal | |
| AS_7-006 | Building Air handling unit Temperature con... | Setp MA energy recovery | 32 °C | normal | |
| AS_7-007 | Building Air handling unit Temperature con... | Outdoor | | normal | |
| AS_7-008 | Building Air handling unit Temperature con... | Condition limit | COFF | normal | |
| AS_7-009 | Building Air handling unit Temperature con... | Outdoor | | normal | |
| AS_7-010 | Building Heat generation Boiler Plant Heat... | Flow temperature setpoint | 70 °C | normal | |
| AS_7-011 | Building Air handling unit | Heat | | normal | |
| AS_7-012 | Building Air handling unit Pressure Sensor | Outdoor heating demand | 70 °C | normal | |
| AS_7-013 | Building Air handling unit | Outdoor | | normal | |
| AS_7-014 | Building Air handling unit | Heating demand | True | normal | |
| AS_7-015 | Building Air handling unit | Outdoor heating demand | 70 °C | normal | |
| AS_7-016 | Building Air handling unit Temperature con... | Heat | | normal | |
| AS_7-017 | Building Air handling unit Temperature con... | Outdoor | | normal | |
| AS_7-018 | Building Air handling unit Temperature con... | Enable function | 21 °C | normal | |
| AS_7-019 | Building Air handling unit Temperature con... | Condition value for room | 21 °C | normal | |
| AS_7-020 | Building Air handling unit Pressure Sensor | Operating state | 0/0 | normal | |
| AS_7-021 | Building Heat generation Boiler Plant Heat... | Cook temp at breakfast | 23.000000/170728 °C | normal | |
| AS_7-022 | Building Air handling unit Temperature con... | Onk input for healthy air | 23.100000/169227 °C | normal | |

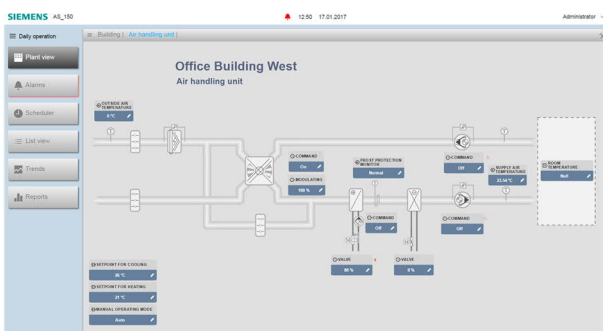
Heating curve

- Heating curve as a graphical background image with the most important operating parameters.
- List displays for complete operation of all heating curve parameters.



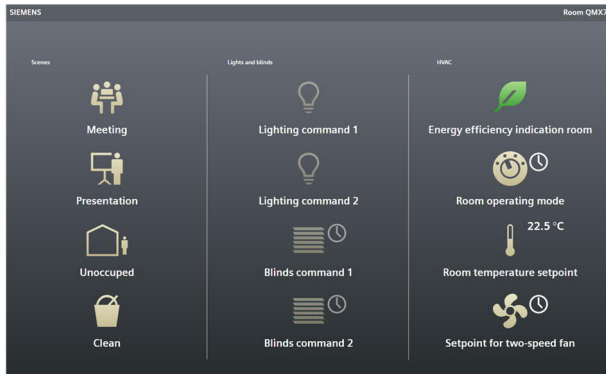
Plant graphics

- Animate 2D and 2D+ symbol.
- Look & feel follows Designo CC style.



End user room operation

- Operate lighting, shading, and HVAC components.
- User interface optimized for the end user to efficiently operate office, meeting, conference rooms, etc.
- Standard templates for meeting and office rooms.



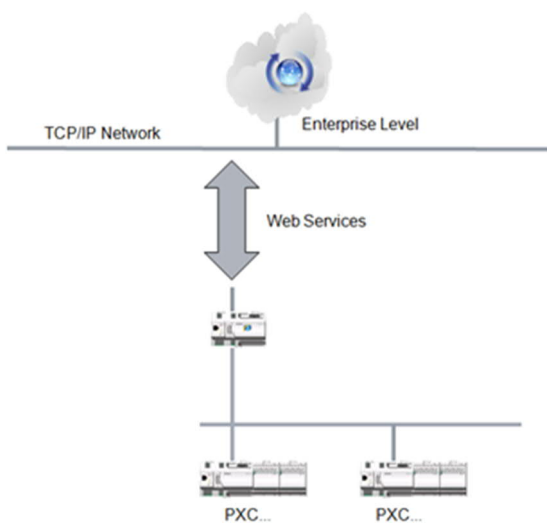
Energy dashboards

- Standard template to display and compare energy consumption.
- Comprehensive selection of graphic, configurable controls.



Northbound integration

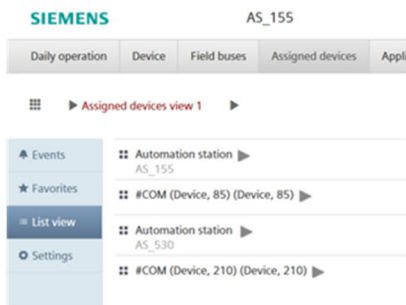
- External access to IT application on BACnet objects of a Desigo system via Haystack tagging or Haystack REST API.
- Project Haystack is an initiative to simplify handling data from the Internet of Things (IoT) and to optimize it for building automation and control (<http://project-haystack.org/>).
- Application examples: Data access to third-party devices with HTML5.0 Browser, customized apps, SAP systems, etc.



Commissioning and service

- Integrated commissioning on all PXM touch panels and PXG3 web interfaces.
- No tool for commissioning required. Commissioning via standard HTML5.0 compatible web browser or directly on the touch panel.
- Online device discovery and data refresh.

- Generic operation of all objects and object properties of assigned devices.
- Workflows are the same as the introduced room automation stations (e.g., firmware update or upload of configuration data).



Engineering

- Graphics editor is integrated in ABT Site.
- Flexible and efficient creation of customized user interface: Page layout, navigation, texts, colors, operating elements (list, buttons, hyperlinks, etc.).
- Efficiently take advantage of existing graphics.
- Simulated preview of graphics in the tool during creation.
- All functions available for editing and creating graphics via a standard web browser, no additional tool required.
- Graphics library with a large selection of symbols and templates:
 - 2D and 2D+ symbols for plant graphics
 - Templates for technical primary plants and rooms, meeting and office environments, energy dashboards
 - RC-specific localization possible



For more information about Desigo Control Point, see:

- *Desigo Control Point Planning and Installation Manual (A6V11170804)*
- *Desigo Control Point Engineering Manual (A6V11211560)*
- *Desigo Control Point Operation Manual (A6V11211557)*

18 Automation stations

The Desigo PX range is based on freely programmable automation stations. They provide the infrastructure to accommodate and process system-specific and application-specific functions. The PX range of automation stations comprises the compact and modular series.

See *Desigo PX - Automation system for HVAC and building services - System overview* (CM110756).

See *Automation stations modular series PXC..D, PXC..-E.D, PXA40-..* (CM1N9222).

See *Automation stations compact model PXC..D* (CM1N9215).

Control Functions

The D-MAP programming language lets you program and parameterize plants, using function blocks and compounds. The graphics-based data-flow programming in Xworks Plus (XWP) lets you implement all the necessary control strategies for optimum operation.

System Functions

The distributed functions, which ensure the overall functioning and inter-operation of all plants, are described in the following chapters and documents:

- For alarm strategy, see chapter *Alarm Management*.
- For time scheduling, see chapter *Calendars and Schedulers*.
- For access rights and user designations, see *IT Security in Desigo Installations* (CM110663).
- For emergency operation and forced control, see chapter *Control Concept*.
- For wiring tests with Desigo Point Test Tool, see chapter *Desigo Workflow, Tools and Programming*.

Cyclical Processing

One PX automation station contains one downloaded D-MAP program. A D-MAP program cannot run on two automation stations, that is, there are no overlapping programs across automation stations. A downloaded D-MAP program does not run automatically. It must be started explicitly and is executed in accordance with the cyclical processing principle, that is, all D-MAP blocks in an automation station are processed in a repeating cycle.

Cycle time

A minimum and maximum cycle time is defined for each automation station. If the processing of all blocks is:

- Shorter than the minimum cycle time, the next processing cycle is delayed until the minimum cycle time has elapsed.
- Longer than the maximum cycle time, the next processing cycle starts as soon as possible.

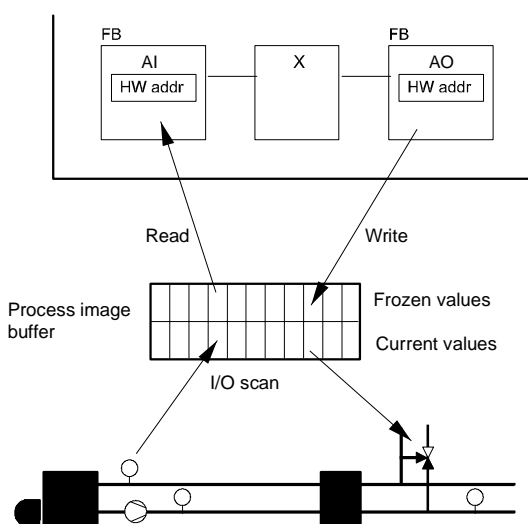
The processing order of the individual blocks:

- Does not depend on their arrangement on the plan (D-MAP program)
- Can be set explicitly when creating the D-MAP program

Process image

The values at the physical inputs and outputs are displayed in the automation station via the process image. There are two instances of the process image:

- The frozen process image does not change during a processing cycle. D MAP programs only read from or write to this instance of the process image.
- The active process image is continuously connected to the real plant.



Values read in cycle 1 are processed in cycle 2. Output values calculated in cycle 1 are transferred to the peripherals in cycle 2.

18.1 Device object

Each automation station contains a device object. The device object:

- Contains the device and system information for the automation station
- Is based on the standard BACnet object as defined in the BACnet standard, and contains additional proprietary properties
- Is always present and is set up in the automation station with initial values
- Is not programmed in the CFC Editor as a function block and is not loaded with the program

You can monitor property values through a BACnet client (e.g., Desigo CC, XWP). You can change default values. You cannot read changed values back into Xworks Plus (XWP). When an automation station is replaced, you must reenter any changes made to property values.

| Property | Value |
|---------------------------|--|
| Object name | TISite'AS02 |
| Object identifier | 0x2213402 |
| System status | Operational |
| State flag | 1100 = 'In alarm' 'Fault' |
| Model name | PXC100-E.D / HW=V3.00 |
| Firmware revision | FW=V6.00.018 / SBC=11.01 / FLI=06.00 / BBI=13.06 / LWI=01.10 / PMC=01.30 / STF=01.20 |
| Serial number | SN=150120C61487 |
| Application SW version | Appl_SW_Vers |
| Location | |
| Local date | 2/23/2015 |
| Local time | 12:45:29 PM |
| DST status | No |
| DST start date | 03.06.Sunday |
| DST start time | 2:00:00 AM |
| DST end date | 10.06.Sunday |
| DST end time | 3:00:00 AM |
| UTC offset | -60 |
| Minimum cycle time | 250 |
| Maximum cycle time | 1000 |
| Program modification time | 2/17/2015 12:59:38 PM |
| Protocol version | 1 |
| Protocol revision | 12 |
| I/O statistics | |
| Database revision | 1 |
| Last restore time | |
| Timeout for backup | 00:01:00 |

Object name
Id=77 Object name

The serial number in the row *Serial number SN=150120C61487* consists of:

- 15 = Year
- 01 = Month
- 20 = Day
- C = Hardware version
- 61487 = Consecutive number

Division into groups

The properties of the device object can be divided into groups based on category, e.g.:

- BACnet communication and BACnet interoperability
- Global properties and system functions
- Local functions and settings
- Statistics and diagnostics

Properties for BACnet communication and interoperability

These properties ensure communication and interoperability between BACnet devices, and are specified in the BACnet standard, e.g., Protocol version [ProtVn] and Vendor name [VndrNam]. Individual properties such as Object identifier [ObjId] are set up by XWP during the commissioning process on the network side.

Global properties

Individual properties of the device object are defined as global properties, because, from the system perspective, all automation stations on one site must have the same value. Global properties are only adjustable on the primary server.

Local properties

The device object contains local properties, which are necessary for the parameterizing and functional scope of global objects and for functions, such as life check, time synchronization and the replication of global objects. Local properties also include properties for the system status of the automation station, the time stamp for the generation of the program and the setting of the buffer size of the alarm queue.

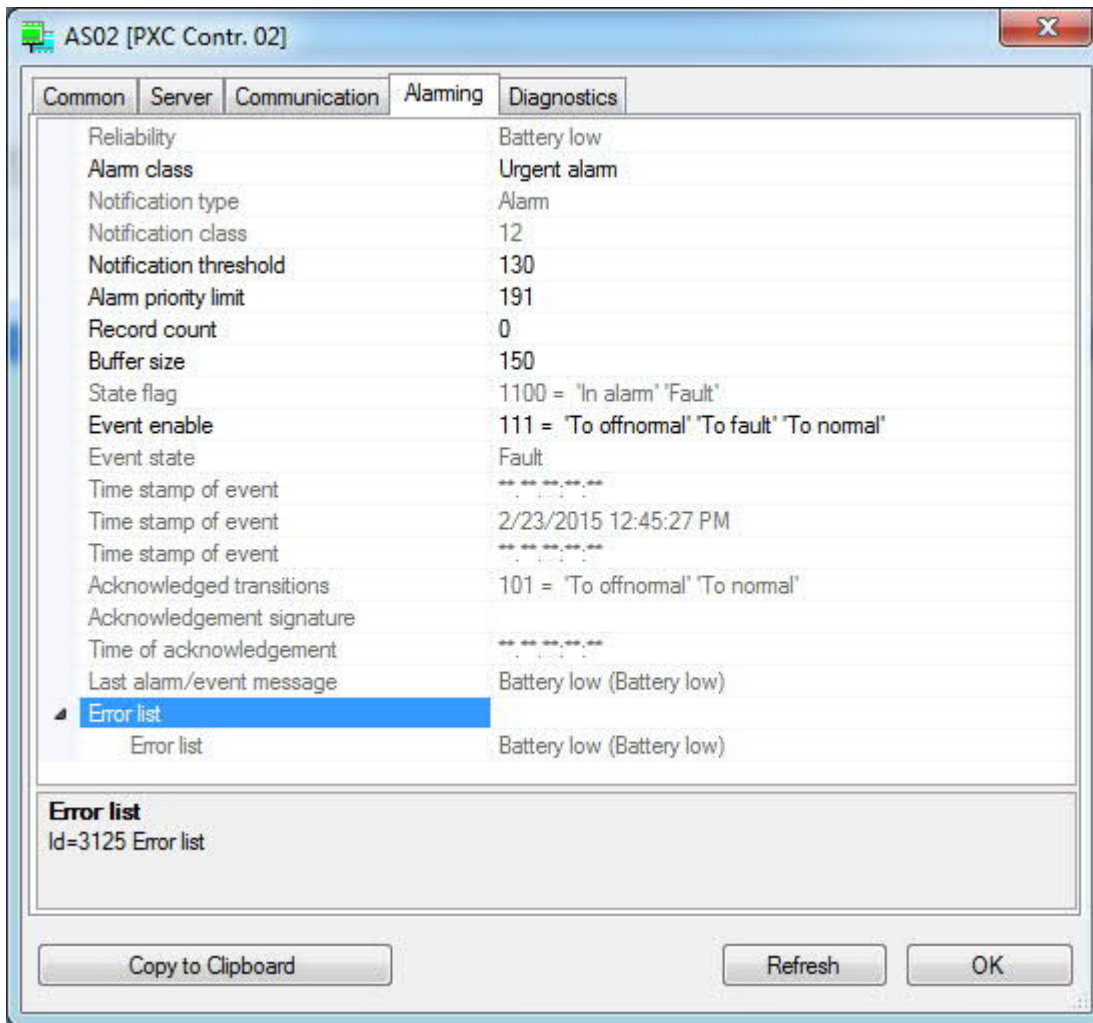
These properties can be reviewed in the **Online Properties** window in the Network Configurator or CFC Editor in XWP.

Properties for statistics and diagnostics

These properties contain statistical and diagnostic information and can be reviewed in the **Online Properties** window in the Network Configurator or CFC Editor in XWP.

18.2 Device info object

The Device Info Object is a proprietary BACnet object and contains the alarming function for the automation station.



Properties for system alarms and system events

The device object has an alarm mechanism, because system alarms and system events, which cannot be assigned to a data point, may occur in an automation station. The alarm state machine and alarm-relevant connections are mapped to the BACnet properties of the device object.

18.3 Error sources and monitoring functions

There are various error sources, e.g.:

| Error | Effect |
|---|--|
| Memory error, e.g., faulty flash memory | Desigo PX stops working. |
| Battery failure | Desigo PX continues working. |
| Failure of backup server recognized by primary server | Desigo PX recognizes the fault and transmits the relevant alarm. |

Non-critical errors / configuration errors

Non-critical hardware and software errors are identified by Desigo PX and registered as a device object alarm.

Critical errors

When a critical hardware or software error occurs, the automation station tries to restart. If the same error is detected three times within 15 minutes, the automation station switches to the COMA operating state. If the **Fault LED** is lit, the automation station is in the COMA operating state.

Online properties for diagnostics

The values in the **Online Properties** window in Xworks Plus (XWP) give clues about the operation of the automation station.

The screenshot shows a software window titled "AS02 [PXC Contr. 02]" with a "Diagnostics" tab selected. The window displays a list of system statistics and a detailed log of events.

| Category | Value |
|-----------------------------------|---|
| Last restart reason | Warmstart |
| Time of device restart | 2/23/2015 12:45:14 PM |
| Cycle time statistics | |
| Memory statistics | |
| Diagnostics | |
| 23.02.2015 12:45:28 | Communication is CONFIGURED,CCnfConfigurationMgr.cpp(923) |
| 23.02.2015 12:45:28 | Ethernet Link is UP, IP Address: 10.169.22.218,CCnfConfigurationMgr.cpp(1593) |
| 23.02.2015 12:45:15 | Normal Start, No Coldstart Reason, Network-Forced Reset |
| 23.02.2015 12:44:42 | ResetUserProgram done,Dms_InterfCDmsCtrl.cpp(451) |
| 23.02.2015 12:38:09 | Communication is CONFIGURED,CCnfConfigurationMgr.cpp(923) |
| 23.02.2015 12:38:09 | Ethernet Link is UP, IP Address: 10.169.22.218,CCnfConfigurationMgr.cpp(1593) |
| 23.02.2015 12:38:08 | Ethernet Link is UP, IP Address: 172.16.87.12,CCnfConfigurationMgr.cpp(1593) |
| 23.02.2015 12:37:59 | Normal Start, No Coldstart Reason, Network-Forced Reset |
| 23.02.2015 12:37:08 | Ethernet Link is UP, IP Address: 172.16.87.12,CCnfConfigurationMgr.cpp(1593) |
| 23.02.2015 12:37:08 | Ethernet Link is UP, IP Address: 172.16.87.12,CCnfConfigurationMgr.cpp(1593) |
| 23.02.2015 12:16:55 | Normal Start, No Coldstart Reason, HW-Forced Reset |
| Modem statistics | < reset statistics > |
| Communication statistics | < reset statistics > |
| Startup statistics | |
| Normal Start | 3 |
| Application Stopped Start | 0 |
| Battery Failed | 0 |
| RAM Pattern Failed | 0 |
| HW-Forced | 0 |
| SW-Forced | 0 |
| HW-Forced Reset | 1 |
| SW-Forced Reset | 0 |
| Network-Forced Reset | 2 |
| SW-Watchdog Reset | 0 |
| Runtime Error | 0 |
| Power Fail | 0 |
| Assertion Failed | 0 |
| Exception Bus Error | 0 |
| Exception Address Error | 0 |
| Exception Illegal Instruction | 0 |
| Exception Zero Divide | 0 |
| Exception Check Instruction | 0 |
| Exception Overflow Error | 0 |
| Exception Privilege Violation | 0 |
| Exception Uninitialized Interrupt | 0 |
| Exception Spurious Interrupt | 0 |
| Exception Unused Trap | 0 |
| Exception Unknown Exception | 0 |
| Serial line statistics | < no Entry > |

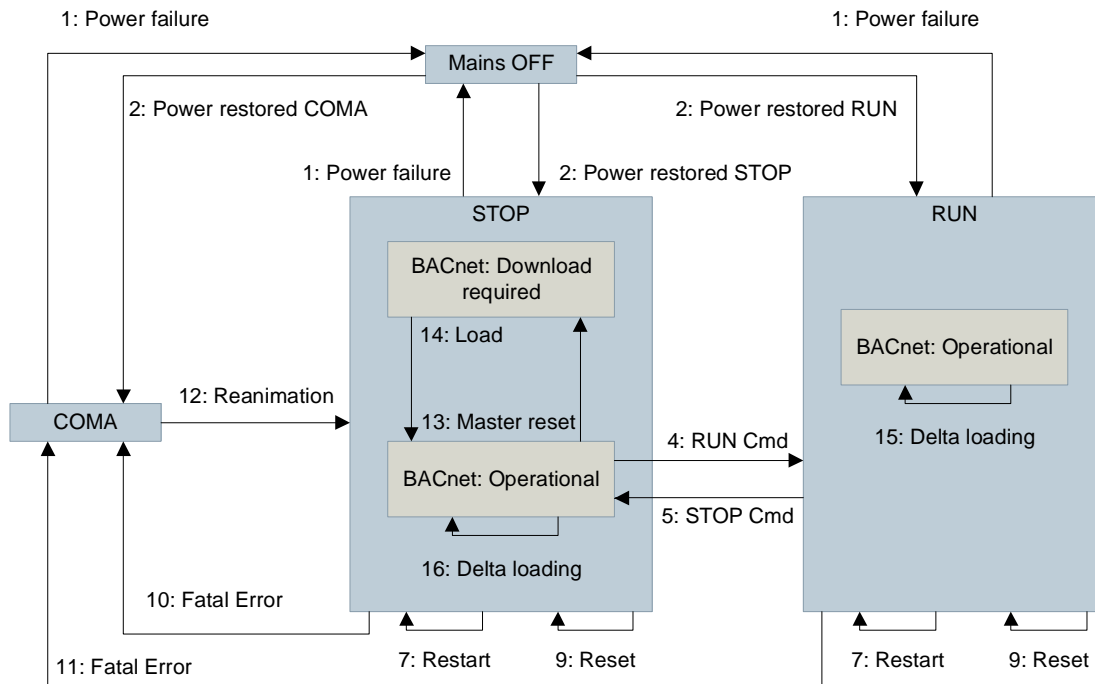
At the bottom of the window, there is a "Diagnostics" section with "Id=3041 Diagnostics". Below this are three buttons: "Copy to Clipboard", "Refresh", and "OK".

18.4 Operating states

A PX automation station has the following operating states:

- **STOP:** The D-MAP program is stopped.
- **RUN:** The D-MAP program runs.
- **KOMA:** The automation station is in a prolonged sleep mode.

The following figure shows the operating states and the associated transitions:



Operating states

Mains off

- No power supply

STOP

- I/O scan active
- Ready for wiring test (only possible without D-MAP program loaded) (PXM20, for PTM modules only)
- D-MAP program processing stopped
- Communication with XWP: Master reset, complete loading and delta loading allowed
- BACnet communication with Desigo CC and PXM20: ReadProperty, WriteProperty, Who-Has, COVs, EventNotification, AcknowledgeAlarm GetEventInformation
- COVs: For values changed by the operator, values cannot be changed by the program
- Alarming: Alarm monitoring inactive, no new alarms or events are generated (the device info object can still generate alarms and system events). Notification of saved alarms and events is possible if recipient lists are set up. GetEventInformation and AcknowledgeAlarm possible.
- Primary server in the STOP state: Primary server is not active, that is, no life check, no time synchronization and no replication of global objects
- Backup server in the STOP state: The backup server is not active, that is, no time synchronization and no replication of global objects by primary server. The backup server will not accept changes to global objects by a client.

RUN

- I/O scan active
- Wiring test not allowed
- D-MAP program processing active
- Communication with XWP: Master reset and complete loading not allowed, delta loading allowed
- BACnet communication with Desigo CC and PXM20: ReadProperty, WriteProperty, Who-Has, COVs, EventNotification, AcknowledgeAlarm GetEventInformation.
- COVs: For values changed by the program and operator
- Alarming: Alarm monitoring active, notification of alarms and events, GetEventInformation and AcknowledgeAlarm

- Primary server in the RUN state: Primary server is active, that is, life check, time synchronization and replication of global objects
- Backup server in RUN state: The backup server is active, that is, time synchronization and replication of global objects by primary server. The backup server does not accept changes of global objects by a client.

COMA

- I/O scan not active
- Communication with XWP not active
- BACnet communication not active
- Wiring test not possible
- D-MAP program processing stopped

Transitions

1 Power failure

Power failure

2 Power restoration STOP

Power restoration. Operating state before power failure was STOP.

Actions (cold start response):

- Cold start I/O scan: Default values for output modules
- Cold-start variable function blocks: Volatile variables are initialized with initial value. Non-volatile variables retain their last value.

The STOP state is reached when the I/O scan is finished.

3 Power restoration RUN

Power restoration. Operational status before power failure was RUN.

Actions (cold start response):

- Cold start I/O scan: Default values for output modules
- Cold-start variable function blocks: Volatile variables are initialized with initial value. Non-volatile variables retain their last value.
- System event: Power restoration.

D-MAP processing starts when the first I/O scan is finished.

4 RUN Cmd

Explicit command via dialog in XWP or BACnet (DeviceObject, Out of service property [OoServ])

Actions (warm start action):

- Implicit warm start I/O scan: I/O scan continues to run
- Implicit warm-start variables function blocks: All variables retain their last value
- System event: Change to operating state

D-MAP processing starts.

5 STOP Cmd

Explicit command via dialog in XWP or BACnet (DeviceObject, Out of service property [OoServ])

Actions:

- System Event: Change to operating state
- Stop D-MAP processing at the end of current cycle

I/O scan continues.

6 Restart

Restart of the automation station due to software error.

Actions (cold start response):

- Cold start I/O scan: Default values for output modules
- Cold start function block variables: Volatile variables are initialized with initial value. Non-volatile variables retain their last value.

The STOP state is reached when the I/O scan is finished.

7 Restart

Restart of the automation station due to software error.

Actions (cold start response):

- Cold start I/O scan: Default values for output modules
- Cold start function block variables: Volatile variables are initialized with initial value. Non-volatile variables retain their last value.
- System event: Restart

D-MAP processing starts when the first I/O scan is finished.

8 Reset

Explicit reset of automation station via hardware push button.

Actions (cold start response):

- Cold start I/O scan: Default values for output modules
- Cold start function block variables: Volatile variables are initialized with initial value. Non-volatile variables retain their last value.

The STOP state is reached when the I/O scan is finished.

9 Reset

Explicit reset of automation station via hardware switch.

Actions (cold start response):

- Cold start I/O scan: Default values for output modules
- Cold start function block variables: Volatile variables are initialized with initial value. Non-volatile variables retain their last value
- System event: Reset

D-MAP processing starts when the first I/O scan is finished.

10, 11 Fatal Error

Restart due to fatal error in the software or in the D-MAP program. Criterion: Same error occurs three times within 15 minutes.

Actions (cold start response):

- Stop I/O scan If possible: Loss of hardware output values for compact and modular automation stations
- Stop BACnet communication
- Stop XWP communication

D-MAP processing is stopped.

12 Reanimation

Only possible by deleting the D-MAP program (press **ForceFWDownload** pin and reset the automation station).

Actions (cold start response):

- Change of required operating state to STOP
- Cold start I/O scan: Default values for output modules

The STOP state is reached when the I/O scan is finished.

13 Master reset

Deletion of D-MAP program on automation station with XWP.

- Cold start I/O scan: Default values for output modules

D-MAP program data including system and event queue is deleted.

14 Load

Complete loading of a new D-MAP program.

- Before downloading, a master reset must be carried out.

Function block variables are loaded with initialized values.

15 Delta download

D-MAP program changes are loaded.

16 Power restoration - COMA

Power restored. Operating state before power failure was COMA.

Actions (cold start response):

- Stop I/O scan
- Stop BACnet communication
- Stop XWP communication

D-MAP processing is stopped.

Summary

Every time the automation station restarts (Powerfail, Reset) a cold start is carried out.

The operating state is stored as a non-volatile variable.

The operating state is mapped as follows to the system status [SysSta] property of the device object:

| Operating mode | System status property [SysSta] |
|--------------------------------|---------------------------------|
| STOP (no D-MAP program loaded) | DOWNLOAD_REQUIRED |
| STOP (D-MAP program loaded) | NON_OPERATIONAL |
| RUN (D-MAP program loaded) | OPERATIONAL |

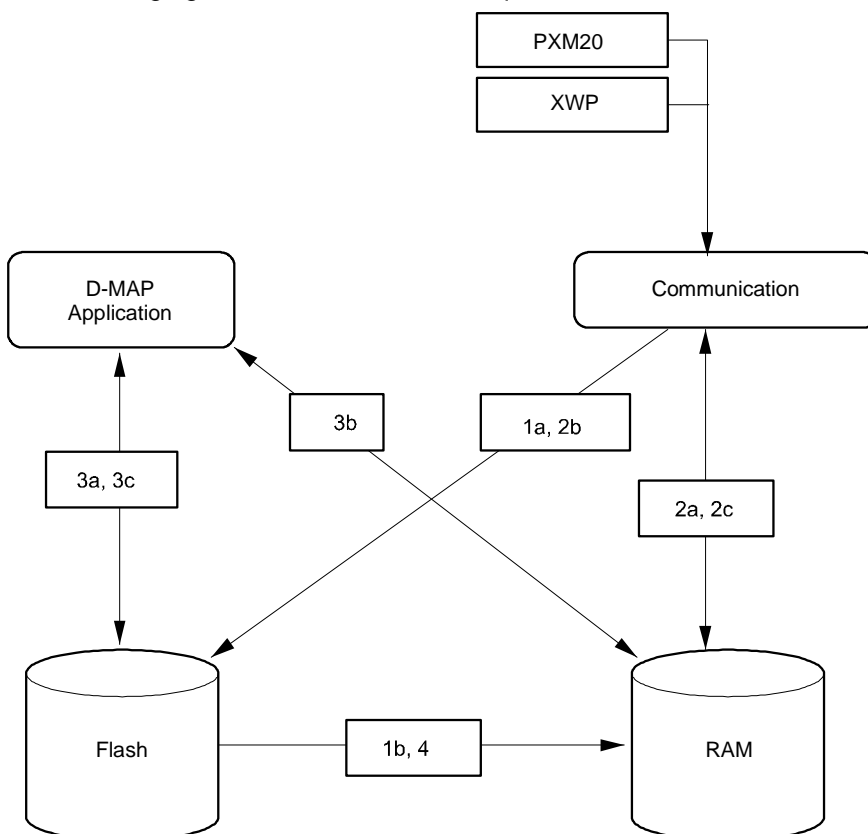
18.5 Data storage

The following memory types are used in the automation station:

- RAM: The content is lost during a cold start. Read and write access is possible any time without any special action.
- Battery supported RAM: Operating hours and trend data are preserved during a cold start if the battery is loaded.
- Flash memory: The content is retained during a cold start. Read access is possible at any time. Write access is only possible via a special driver and with restrictions (access time, sequential only).

The data and code of a D-MAP program are saved in the flash memory during the download process. A copy of the data is always stored in the RAM so that the D-MAP program can access data efficiently for processing purposes. This means, that all changes to the program data must be updated both in the RAM and in the flash memory.

The following figure shows the various sequences:



Downloading the D-MAP program

1. The D-MAP program (code and data blocks) is copied to the flash memory (1a). A copy of the data blocks is created in the RAM (1b) for later modification by the D MAP program.

Read/Write via communication system

2. When writing data, the data is written to the RAM (2a) and the flash memory (2b). Read access to the data is via the RAM (2c).

Processing the D-MAP program

3. The D-MAP program code is read from the flash memory (3a). The program data is modified in the RAM (3b). Non-volatile process variables (e.g., adaptive control parameters, hours run, etc.) are written by the function blocks into the flash memory (3c) at regular intervals (once per day) or saved in the battery supported RAM.

Starting the automation station

4. At each restart of the automation station, a copy of the data (data blocks) is created in the RAM (4) from the flash memory (including all communication and D-MAP changes).

19 Logical I/O blocks

I/O blocks are used to register and transmit raw data to and from the plant, and to convert, process and integrate it into the program.

The following options are supported:

- Raw data from or to the input or output modules.
- Raw data from or to the PPS2 interface (room units) (not for the modular series PXC...D)
- Data referenced via the Technical Designation (TD) and accessed either in the same automation station (without a connection), or peer-to-peer via BACnet services.
- Data made available via a Discipline I/O of a room automation station or third-party device.

The term I/O blocks refers collectively to the individual input blocks and output blocks.

- Input blocks are used to enable an input signal (e.g., a measured value) in the program to be handled as a process value.
- Output blocks are used to enable a process value to be transmitted as an output signal (e.g., a positioning command).

Value blocks act as a link between program pins, and are used to temporarily store a process value, and if necessary, to display it on a client operator station. A special version of the value block, the Value block for operation provides a simplified means of operation from an operator client (without the facility to override values manually).

Counter Input blocks (CI blocks) are used to enable a counter value (e.g., from a gas or electricity meter) to be processed in the application as a real-number process value. In this process, the counter value (pulse) is converted in the block into the associated physical variable.

Integration I/Os (Discipline I/O blocks) are used, e.g., to integrate room automation or third-party devices

| Input blocks | Output blocks | Value blocks | Value blocks for operation |
|-------------------------------|--------------------------------|---------------------|----------------------------|
| Analog Input (AI, AI RED) | Analog Output (AO, AO RED) | Analog Input (AVAL) | Analog Input (AVAL_OP) |
| Binary Input (BI, BI RED) | Binary Output (BO, BO RED) | Binary (BVAL) | Binary (BVAL_OP) |
| Multistate Input (MI, MI RED) | Multistate Output (MO, MO RED) | Multistate (MVAL) | Multistate (MVAL_OP) |
| Counter Input (CI) | | | |
| Accumulator (CI ACC) | | | |
| Discipline I/O | | | |

Program view and system view

I/O blocks are displayed in two different views:

- The program view shows an I/O block with the pins and attributes required for configuration purposes and to create the program. This is the display format used in Xworks Plus (XWP).
- The system view shows the I/O blocks as standard BACnet objects. These BACnet objects and the associated properties are then available to clients from where they can be operated and monitored.

BACnet functions

All the blocks listed above are implemented in accordance with the BACnet standard. Therefore additional functions are available, such as alarm management. These blocks incorporate a mechanism which acts as an alarm source for blocks available as standard BACnet objects in the BACnet network. By use of various BACnet services, a given event is displayed as an alarm event on the relevant clients from where the alarm can be processed, that is, viewed, acknowledged and/or reset.

In XWP these functions can be tracked via the relevant values at the block pins in online test mode.

19.1 General functions

Blocks: AO, BO, MO, AVAL, BVAL, MVAL

This section describes the general functional scope shared by many of the I/O blocks. Each subsection includes a list of the blocks to which that subsection applies. Any block-specific details which are not shared by other blocks are described together with the block concerned.

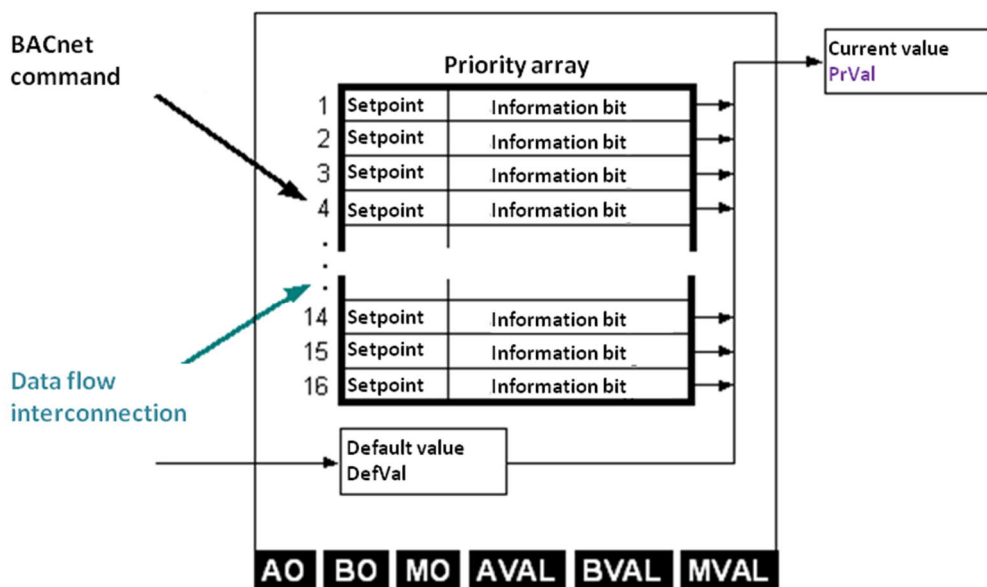
Priority mechanism

Basic function

In order to evaluate the various defined setpoints received from the BACnet command system and via the data flow connections, the AO, BO, MO, AVAL, BVAL and MVAL blocks each incorporate a priority array [PrioArr].

All external sources write their defined setpoint and information bit (enable signal) into this [PrioArr]. The block then evaluates these entries continuously, in order to determine the valid present value [PrVal].

The [PrioArr] holds up to 16 different entries, each consisting of a setpoint definition and the associated information bit (enable signal). The input number also indicates the priority of the entry, where 1 is the highest and 16 the lowest priority. Each priority level has a predefined meaning.



Determining [PrVal]

The block continuously evaluates the valid present value at the output [PrVal]. It selects the value that has the highest priority of those whose information bit (enable signal) is also set. If none of the information bits is set, the default value [DefVal] is processed.

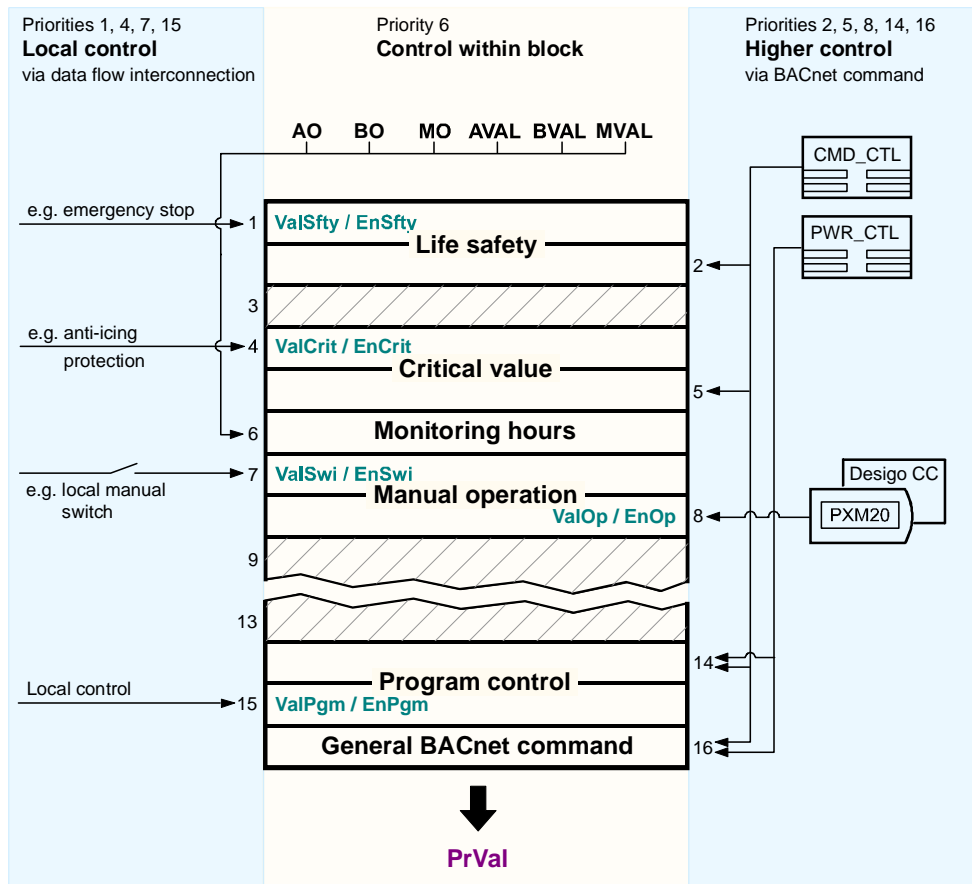
Structure of the Priority Array [PrioArr]

Each priority level has a predefined meaning.

In the [PrioArr], two adjacent priority levels each are reserved for life safety, manual operation and plant operation.

- The higher priority (lower number) of each pair is reserved for local control and monitoring, close to the plant (priority 1, 4, 7 and 15).
- The lower priority (higher number) of each pair is reserved for higher level control and monitoring (priority 2, 5, 8 and 16).
- Priority level 6 is specifically designed for switch-on and switch-off delays and to maintain minimum ON and OFF times.

This ensures that, e.g., an on-site EMERGENCY OFF command, initiated at the plant level, takes priority over a safety function from a higher-level subsystem.



10011Zg2en_01

Priority 6

Priority entry 6 is used to forward the switch commands resulting from [PrioArr] to the [PrVal] output after a delay. This enables you to implement both switch-on and switch-off delays and minimum ON and OFF times. For this purpose, the internal block logic imports the Present value [PrVal] into the priority 6 entry. While the delay times referred to above are running, priority 6 is set to active and so takes priority over priority levels 7...16. Outside these delay times, priority 6 is always inactive.

Locating this function in the [PrioArr] between priorities 1...5 and 7...16 has the following consequences:

- Commands with a priority level of 1...5 are always executed immediately, irrespective of any currently active delay times.
- Commands with a priority level of 7...16 are always overridden by any currently active delay times.

Unlike all the other entries in [PrioArr], the commands and information bit for priority 6 are generated exclusively by the BO, MO, BVAL and MVAL blocks. A priority 6 entry cannot be written from an external source.

The switch-on and switch-off delay

As soon as one of the commands with a priority of 7...16 determines the [PrVal] which will therefore cause the present state of [PrVal] to change, the entry for priority 6 is set up as follows:

If the switch-on delay [DlyOn] or switch-off delay [DlyOff] is greater than 0:

1. Priority 6 adopts the still unchanged present value [PrVal].
2. Priority 6 is set to active.
3. The switch-on or switch-off delay timer starts.
4. After expiry of [TiOnMin] or [TiOffMin], priority 6 is set to inactive.

If the delay times [DlyOn] or [DlyOff] are equal to 0, no action is taken.

If the new value which determines [PrVal] is the same as the current [PrVal], then, here too, no action is taken.

The minimum on/off time

For each change at the output [PrVal] from OFF to Stage n or from Stage n to OFF, the entry for priority 6 is set up as follows:

If the minimum ON-time [TiOnMin] or OFF-time [TiOffMin] is greater than 0:

1. Priority 6 adopts the new present value [PrVal].
2. Priority 6 is set to active.
3. The timer for the minimum on-time or off-time is started.
4. After expiry of [TiOnMin] or [TiOffMin], priority 6 is set to inactive.

If the minimum switch-times [TiOnMin] / [TiOffMin] are set to 0, no action is taken.

Constraints

- The functions described above are supported only by the BO, MO, BVAL and MVAL blocks.
- With multistage switch commands, the monitoring periods are enabled only when switching from OFF to Stage n or from Stage n to OFF. When switching from one stage to another (e.g., Stage 1 to Stage 2), the timer times are not enabled.
- However, any timer times already running will continue to run.

Application

- Unnecessary on/off switching operations can be prevented by activating minimum switch-on or switch-off times.
- Activating switch-on or switch-off delays ensures that run-on time delays are maintained.

Information bit

In order for a given value to be included in the evaluation of [PrioArr], its information bit must be set.

The following applies to priority 1,4,7 and 15 (data flow connection): The relevant information bit is set via pins [EnSfty], [EnCrit], [EnSwi] and [EnPgm].

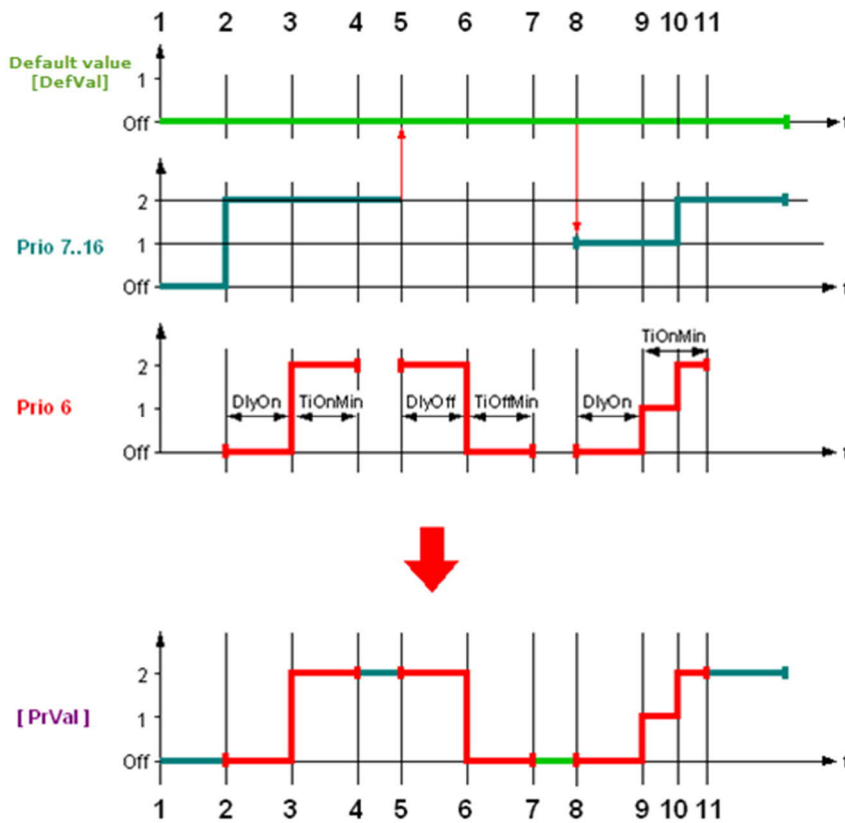
The following applies to priority 2, 5, 8, 14 and 16 (BACnet command system): When a given value is commanded via BACnet, the value concerned is entered in the [PrioArr] and the associated information bit is set automatically.

The following applies to priority level 6: Both the value and the information bit are handled by the block concerned.

| Prio | Meaning | Use | Access via |
|------|--|--|--|
| 1 | Safety value (life safety) Reserved for the initiation of safety functions (1 = highest priority). If priority 1 or 2 becomes the determining value for [PrVal], then the value concerned is transmitted immediately to the [PrVal] output. It is not subject to the delay times defined for priority 6. | Local safety function, e.g.: - Fire - EMERGENCY OFF - Service switch - Gas alarm - Thermal package | Data flow interconnection via pins: [ValSfty] and [EnSfty]. Normally, [ValSfty] is a constant and [EnSfty] can be enabled/disabled. |
| 2 | | Higher-level safety function, e.g.: - Smoke extraction | BACnet command system. Access via the CMD_CTL block. |
| 3 | Not used in Desigo. | | |
| 4 | Critical value (plant protection) Reserved for monitoring critical plant states. If priority 4 or 5 becomes the determining value for [PrVal], then the value concerned is transmitted immediately to the [PrVal] output. It is not subject to the delay times defined for priority 6. | Local monitoring of critical plant states, e.g.: - Frost protection (protection from excess cooling) - Interlock of aggregates - Icing protection | Data flow interconnection via pins: [ValCrit] und [EnCrit]. Normally, [ValCrit] is a constant and [EnCrit] is enabled/disabled. |
| 5 | | Higher-level monitoring of critical plant states: - Frost in ventilation system (close dampers, stop fans, switch pump on and open valve) | BACnet command system. Access via the CMD_CTL block. |
| 6 | Minimum switch-on/off time Prevent unnecessary switching operations. Switch-on/off delay Can be used to ensure that run-on delay times are implemented. | | No access! Commands are only generated internally in the block. The timer periods [TiOnMin], [TiOffMin], [DlyOn] and [DlyOff] can be configured in blocks BO, MO, BVAL and MVAL. |
| 7 | Operating value Reserved for manual operation. | Local manual operation, e.g.: - Manual switch | Data flow interconnection via pins: [ValSwi] und [EnSwi]. |

| Prio | Meaning | Use | Access via |
|--------|---|--|---|
| | | - Mode selector switch | |
| 8 | | Higher-level manual operation, e.g.: - Desigo CC - Operating unit - Web client | BACnet command system. Access via: - Desigo CC - Operating unit - Web client |
| 9...13 | Not used in Desigo. | | |
| 14 | Program value Reserved for normal plant operation with monitoring and control. | Superposed control and monitoring of the plant. | BACnet command system. Access is via blocks: - CMD_CTL - PWR_CTL (if control enable signal = Fixed) |
| 15 | | Local control of plant. | Data flow interconnection via pins: [ValPgm] and [EnPgm]. If the program value is a controller variable, then [EnPgm] = True and [ValPgm] = controller variable. If the program value is not a controller variable, then [EnPgm] = False. |
| 16 | Program value Reserved for general cross-PX commands via BACnet references. | | BACnet command system. Access via blocks: CMD_CTL PWR_CTL (if control enable signal is = Released) Cross-PX via various blocks, e.g., ASCHED, BSCHED, MSCHED (name reference list). |
| 17 | Default value [DefVal] If none of priorities 1...16 is active, then the default value [DefVal] is processed instead. | The influence of [DefVal] depends on the state of the block concerned: Out-of-service [OoSrv=False]: [DefVal], like the values of priorities 7...16, is subject to the delay times of priority 6. [OoSrv=True]: [DefVal] is transmitted immediately to the [PrVal] output. | BACnet command system. Access via: - CFC - PXM20 - Web client |

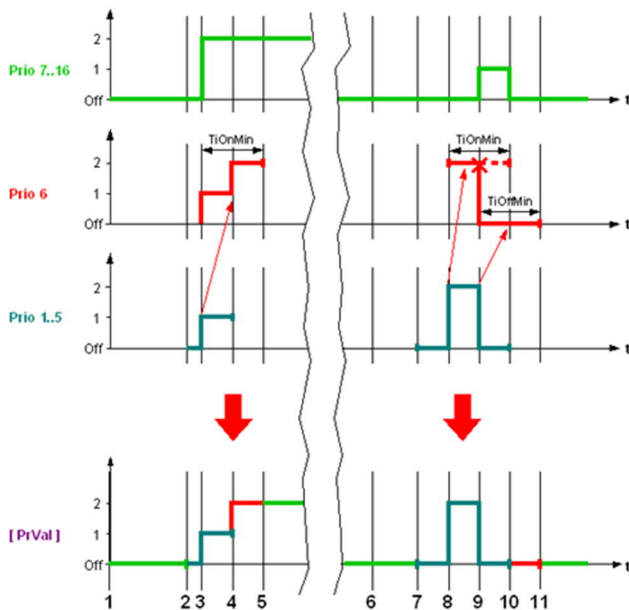
Example: Effect of priorities 7...16 on [PrVal]



| Prio | Use | |
|------|-------------|---|
| 1 | Prio 7...16 | Assumption: The effective switch command from priority (7...16) is <i>Off</i> and is set to active. |
| | Prio 6 | Assumption: Priority 6 is not active. |
| | [PrVal] | Assumption: The [PrVal] output is set to <i>Off</i> . |
| 2 | Prio 7...16 | The effective switch command from priority (7...16) switches from <i>Off</i> to <i>Stage 2</i> . |
| | Prio 6 | Priority 6 adopts the (still unchanged) present value [PrVal= <i>Off</i>] and is set to active. At the same time, the switch-on delay [DlyOn] starts. Throughout the delay time, priority 6 remains active – the associated value remains <i>Off</i> . |
| | [PrVal] | Since priority 6 overrides the effective switch command from priority (7...16), the [PrVal] output remains <i>Off</i> . |
| 3 | Prio 7...16 | n/a |
| | Prio 6 | 1. After expiry of the switch-on delay [DlyOn], priority 6 is released. 2. The effective switch command <i>Stage 2</i> from priority (7...16) is transmitted to the [PrVal] output. 3. Priority 6 adopts the new value of [PrVal] and is set to active again. At the same time, the minimum switch-on time [TiOnMin] is started. Priority 6 remains active throughout this monitoring time. |
| | [PrVal] | The [PrVal] output changes from <i>Off</i> to <i>Stage 2</i> . |
| 4 | Prio 7...16 | n/a |
| | Prio 6 | The minimum switch-on time [TiOnMin] has expired. Priority 6 is released. |
| | [PrVal] | When priority 6 ceases to take effect, the [PrVal] output is once again determined by the effective switch command from priority (7...16). [PrVal] remains at <i>Stage 2</i> . |
| 5 | Prio 7...16 | None of the information bits for priorities (7...16) is active. The resulting switch command is therefore determined by the default value [DefVal]. |
| | Prio 6 | The block starts the switch-off delay [DlyOff]. Throughout this monitoring time, priority 6 is set to active – the associated value remains at <i>Stage 2</i> . |
| | [PrVal] | Since priority 6 overrides the effective switch command [DefVal], the [PrVal] output remains at <i>Stage 2</i> . |
| 6 | Prio 7...16 | n/a |

| Prio | Use | |
|------|-------------|--|
| | Prio 6 | 1. After expiry of the switch-off delay [DlyOff], priority 6 is released. 2. The effective switch command <i>Off</i> from [DefVal] is transmitted to [PrVal]. 3. Priority 6 adopts the new value of [PrVal] and is set to active again. At the same time, the minimum switch-off time [TiOffMin] is started. Priority 6 remains active throughout this monitoring time. |
| | [PrVal] | The [PrVal] output changes from <i>Stage 2</i> to <i>Off</i> . |
| 7 | Prio 7...16 | n/a |
| | Prio 6 | The minimum switch-off time [TiOffMin] has expired. Priority 6 is released. |
| | [PrVal] | Since neither priority 6 nor any of the information bits for priority entries (7...16) is active, the effective switch command is determined by [DefVal]. The output value [PrVal] remains at <i>Off</i> . |
| 8 | Prio 7...16 | At least one of the information bits for priorities (7...16) is active again. The effective switch command from priority (7...16) is <i>Stage 1</i> . |
| | Prio 6 | Priority 6 adopts the (still unchanged) present value [PrVal=Off] and is set to active. At the same time, the switch-on delay [DlyOn] starts. Throughout the delay time, priority 6 remains active – the associated value remains <i>Off</i> . |
| | [PrVal] | Since priority 6 overrides the effective switch command from priority (7...16), the [PrVal] output remains <i>Off</i> . |
| 9 | Prio 7...16 | n/a |
| | Prio 6 | 1. After expiry of the switch-on delay [DlyOn], priority 6 is released. 2. The effective switch command <i>Stage 1</i> from priority (7...16) is transmitted to the [PrVal] output. 3. Priority 6 adopts the new value of [PrVal] and is set to active again. At the same time, the minimum switch-on time [TiOnMin] is started. Priority 6 remains active throughout this monitoring time. |
| | [PrVal] | The [PrVal] output changes from <i>Off</i> to <i>Stage 1</i> . |
| 10 | Prio 7...16 | The effective switch command from priority (7...16) switches from <i>Stage 1</i> to <i>Stage 2</i> . |
| | Prio 6 | The minimum switch-on time [TiOnMin] is still active. Changeover from <i>Stage m</i> to <i>Stage n</i> . With multistage switch commands, the monitoring periods are enabled only when switching from <i>OFF</i> to <i>Stage n</i> or from <i>Stage n</i> to <i>OFF</i> . When switching from one stage to another (e.g., <i>Stage 1</i> to <i>Stage 2</i>), the monitoring periods are not enabled. However, monitoring periods which have already started remain active – priority 6 adopts the new target value. |
| | [PrVal] | A change from <i>Stage 1</i> to <i>Stage 2</i> would not activate priority 6. However, since the minimum switch-on time [TiOnMin] is still active, priority 6 still overrides the effective switch command from priority (7...16). The [PrVal] output changes from <i>Stage 1</i> to <i>Stage 2</i> . |
| 11 | Prio 7...16 | n/a |
| | Prio 6 | The minimum switch-on time [TiOnMin] has expired. Priority 6 is released. |
| | [PrVal] | When priority 6 ceases to take effect, the [PrVal] output is once again determined by the effective switch command from priority (7...16). The [PrVal] output remains at <i>Stage 2</i> . |

Example: Effect of priorities 1...5 on [PrVal]



| Prio | Use | |
|------|------------|--|
| 1 | Prio 1...5 | Assumption: All information bits for priorities 1...5 are inactive. |
| | Prio 6 | Assumption: Priority 6 is not active. |
| | [PrVal] | Assumption: The [PrVal] output is set to <i>Off</i> . |
| 2 | Prio 1...5 | At least one of the information bits for priorities (1...5) is active again. The effective switch command from priority (1...5) is <i>Off</i> . |
| | Prio 6 | Since the effective switch command for priority (1...5) does not cause a change in the [PrVal] output, priority 6 remains inactive. |
| | [PrVal] | The output value [PrVal] remains at <i>Off</i> . |
| 3 | Prio 1...5 | The effective switch command from priority (1...5) switches from <i>Off</i> to <i>Stage 1</i> . |
| | Prio 6 | Priority 6 adopts the new present value [PrVal=Stage 1] and is set to active. At the same time, the minimum switch-on time [TiOnMin] starts without waiting for the delay time [DlyOn]. Note: Entries for priorities (1...5) initialize only the minimum switch-on or switch-off times [TiOnMin] and [TiOffMin] respectively, but not the switch-on and switch-off delays. [TiOnMin] and [TiOffMin] times for which the timer has already started only take effect when all priorities (1...5) are inactive, that is, when the [PrVal] will be determined by one of priorities (7...16). |
| | [PrVal] | Priorities 1...5 are reserved to implement safety functions, and are executed immediately, irrespective of any priority 6 monitoring periods which may already be running. The [PrVal] output is switched immediately from <i>Off</i> to <i>Stage 1</i> . |
| 4 | Prio 1...5 | None of the information bits for priority entries (1...5) is active. |
| | Prio 6 | The minimum switch-on time [TiOnMin] is still active. Priority 6 adopts the new target value from priority (7...16). |
| | [PrVal] | The effective switch command is determined from priority 6. The [PrVal] output changes from <i>Stage 1</i> to <i>Stage 2</i> . |
| 5 | Prio 1...5 | n/a |
| | Prio 6 | The minimum switch-on time [TiOnMin] has expired. Priority 6 is released. |
| | [PrVal] | Since neither priority 6 nor any entries for priorities (1...5) are active, the output [PrVal] is now again determined by the effective switch command from priorities (7...16). The [PrVal] output remains at <i>Stage 2</i> . Note: Switching from <i>Stage 1</i> to <i>Stage 2</i> does not re-start the minimum switch-on time [TiOnMin]. |
| 6 | Prio 1...5 | Assumption: All information bits for priorities 1...5 are inactive. |
| | Prio 6 | Assumption: Priority 6 is not active. |
| | [PrVal] | Assumption: The [PrVal] output is set to <i>Off</i> . |
| 7 | Prio 1...5 | At least one of the information bits for priorities (1...5) is active again. The effective switch command from |

| Prio | Use | |
|---------|--|---|
| | priority (1...5) is <i>Off</i> . | |
| Prio 6 | Since the effective switch command for priority (1...5) does not cause a change in the [PrVal] output, priority 6 remains inactive. | |
| [PrVal] | The output value [PrVal] remains at <i>Off</i> . | |
| 8 | Prio 1...5 | The effective switch command from priority (1...5) switches from <i>Off</i> to <i>Stage 2</i> . |
| Prio 6 | Priority 6 adopts the new present value, [PrVal=Stage 2] and is set to active. At the same time, the minimum switch-on time [TiOnMin] starts without waiting for the delay time [DlyOn]. Note: Entries for priorities (1...5) initialize only the minimum switch-on or switch-off times [TiOnMin] and [TiOffMin] respectively, but not the switch-on and switch-off delays. [TiOnMin] and [TiOffMin] times for which the timer is already running only take effect when all priorities (1...5) are inactive, that is, when the [PrVal] will be determined by one of priorities (7...16). | |
| [PrVal] | Priorities 1...5 are reserved to implement safety functions, and are executed immediately, irrespective of the switch state and of any priority 6 monitoring periods which may already be running. The [PrVal] output is switched immediately from <i>Off</i> to <i>Stage 2</i> . | |
| 9 | Prio 1...5 | The effective switch command from priority (1...5) switches from <i>Stage 2</i> to <i>Off</i> . |
| Prio 6 | Priority 6 adopts the new present value [PrVal=Off]. The still-running minimum switch-on time [TiOnMin] is cancelled. The block re-starts the minimum switch-off time [TiOffMin]. | |
| [PrVal] | Priorities 1...5 are reserved to implement safety functions, and are executed immediately, irrespective of the switch state and of any priority 6 monitoring periods which may already be running. The [PrVal] output is switched immediately from <i>Stage 2</i> to <i>Off</i> . | |
| 10 | Prio 1...5 | All information bits for priorities 1...5 are inactive. |
| Prio 6 | The minimum switch-off time [TiOffMin] is still active. | |
| [PrVal] | The effective switch command is determined from priority 6. The output value [PrVal] remains at <i>Off</i> . | |
| 11 | Prio 1...5 | n/a |
| Prio 6 | The minimum switch-off time [TiOffMin] has expired. Priority 6 is released. | |
| [PrVal] | Since neither priority 6 nor any entries for priorities (1...5) are active, the output [PrVal] is now again determined by the effective switch command from priorities (7...16). The output value [PrVal] remains at <i>Off</i> . | |

Switch types [SwiKind]

Blocks: BO, MO, BVAL, MVAL

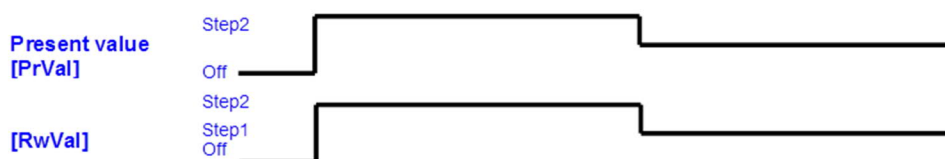
All switching I/O blocks have a configurable switching response. The switching response determines the functioning of the block. The switching functions are subject to the priority mechanism in the [PrioArr] and the switch command delay.

- *Normal*: Direct switching in stages taking into account runtimes (e.g., motors, burners, dampers, etc.).
- *Motor*: Switching in stages for rotating aggregates taking into account ramp-up and ramp-down times (fan-belt protection).
- *Trigger*: Event-driven switching, *last command takes precedence*; integration of a data point (EIB, LONMARK)
- *Switch*: Generation of an ON/OFF pulse of a defined duration.
- *Push button with delay*: Generation of an ON/OFF pulse of a defined duration. The pulse can be extended whenever required.
- *Release (Release Command)*: Issuance of a subsystem-specific release value instead of Present_Value (=Relinquish_Default), if no priority is active in the output object.

| [SwiKind] | BO | MO | BVal | MVal |
|-----------------------|----|----|------|------|
| Normal | • | • | • | • |
| Motor | | • | | |
| Trigger | • | • | | |
| Switch | • | | • | |
| Pushbutton with delay | • | | • | |
| Release | | • | | |

Normal

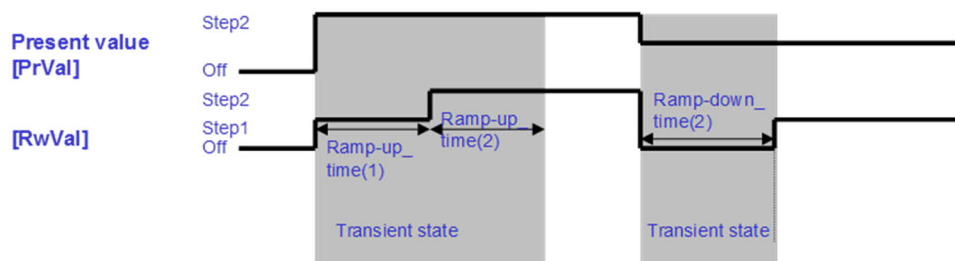
Normal handling of the process values in the [PrioArr]. The configured runtimes are active. The outputs can be switched directly or in stages.



Motor

The *Motor* setting is used when there is a need to allow for ramp-up and ramp-down times due to a rotating centrifugal mass. The programmed times in this setting can be used, e.g., to avoid overloading the fan belt when starting a fan motor.

When the motor is switched down, the system checks on the basis of the ramp-up time whether or not the current motor speed has been reached. The switch-down command is not executed until the motor speed is stable. During the ramp-down period, the effective command to the hardware is *Off*. When the ramp-down time has elapsed, the new command is transmitted to the hardware.



Trigger

In the *Trigger* setting, the source of the last command takes precedence. The valid value is written from the [PrioArr] to the [DefVal] and transmitted to the output. The priority is then released again.

In this setting, Priorities 7...16 are treated equally; Priorities 1...5 have a blocking effect.

The trigger function is used, e.g., for the integration of LON data points. Owing to the event mechanism, this function is not used for P-bus objects.

Switch

The *Switch* setting is used to generate an ON or OFF pulse of a predefined duration. A command via BACnet, or the activation of an Enable signal in one of Priorities 7...16 via the data flow connection initiates an associated pulse (event). The minimum switch-on time [TiOnMin] and/or minimum switch-off time [TiOffMin] must be set. Setting both times can prevent fast switching operations. Priorities 1...5 have a blocking effect.

Pushbutton with delay (time extension)

The *Pushbutton with delay* function is like the *Switch* function, except an active pulse can be extended by another pulse at any time.

Runtimes and monitoring periods

The I/O function blocks are designed for the runtimes and monitoring periods required in HVAC engineering, and can therefore be used directly as components (motors, dampers, fans, etc.).

Different runtimes and monitoring periods can be set, depending on the function concerned.

Runtimes:

- Switch-on/off delay
- Minimum switch-on/off time
- Ramp-up/-down time

Monitoring periods:

- Feedback time with switch-on/off
- Feedback signal deviation during operation

Runtimes

Switch-on/off delay

Blocks: BO, MO, BVAL, MVAL

The switch-on/off delay when applied to the switching I/O blocks causes a delayed output if the switch command was written via Priority 7...16. The delay time affects Priority 6 as described. Switch commands via Priorities 1...5 are executed without a delay.

Minimum switch-on/off time

When applied to the switching I/O blocks, the minimum switch-on/switch-off time causes the output to be blocked for a period of time if the switch command was written via Priority 7...16. The minimum switch-on/off time affects Priority 6 as already described in Section 24.2.1.3. However, switch commands via Priorities 1...5 are executed immediately irrespective of the minimum switch-on/off time.

Ramp-up/down time

The ramp-up/down times (run-up/-down times) can be defined in a table for each stage. These times apply to the two switch types [SwiKind] *Normal* and *Motor*.

The ramp-up time is the time taken by a motor when changing from a lower speed to the next higher speed, to reach the new speed. This limits the current consumption of the motor.

The ramp-down time is the time taken by the motor when switching down from a higher speed, to reach the lower speed. This prevents feedback to the mains supply network and protects the fan belt and the motor.

As a rule, the ramp-up and ramp-down times depend on the centrifugal mass involved, and must be determined separately for each project.

Especially with single-speed motors, the times can be used as Open/Close runtimes (e.g., damper actuator from 0...100%). A moving damper can thus be mapped in the system and the transition signal can, if required, be used for control purposes.

Monitoring periods

Feedback monitoring / process value monitoring

Blocks: BI, MI, BO, MO, BVAL, MVAL

The I/O objects have a monitoring function. The output objects monitor the feedback signal from the plant. For this purpose, an address string must be entered for the [FbAddr] feedback parameter [FbAddr] and the alarm function must be enabled.

The input and value objects can monitor reference values. For this purpose, the relevant reference values must be configured and the alarm function must be enabled.

Deviation monitoring

If the feedback value deviates from the output value [PrVal], a deviation alarm is generated after a configurable time period, and the block status changes to *In Alarm*. When the two values match again, and the configured time period has expired, the alarm and status are reset. There is otherwise no automatic block reaction, that is, if a switch response in the plant is required as a reaction to this alarm, this response must be programmed in CFC via the Disturbance output [Dstb].

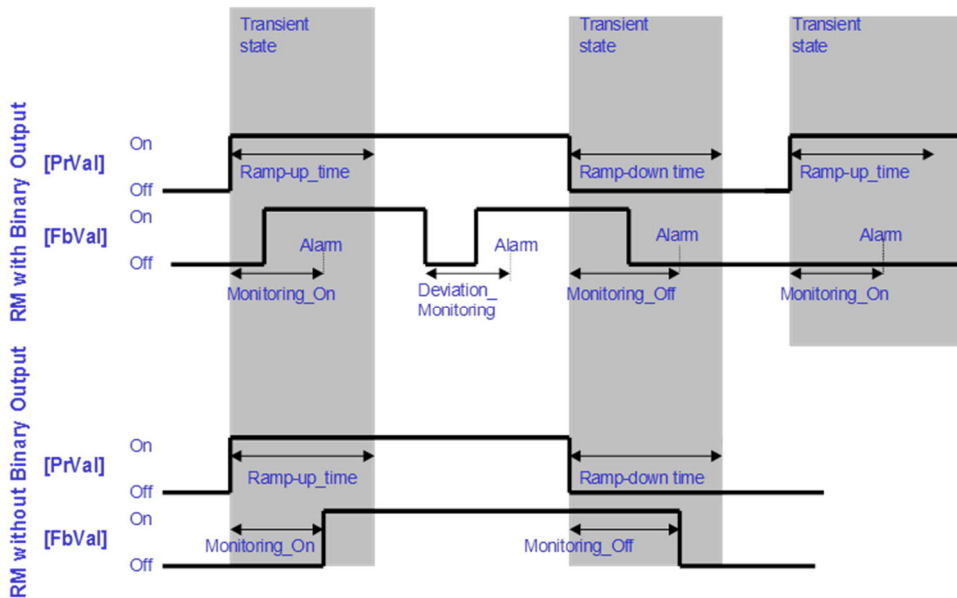
Switch-on/off feedback monitoring

It is also possible to configure the time period during which the maximum deviation of the feedback signal may occur after a switch-on/off operation. If the deviation persists after the monitoring time has expired, an alarm is generated and the status of the block changes to *In alarm*. When the two values match again, and the

configured time period has expired, the alarm and status are reset. There is otherwise no automatic block reaction, that is, if a switch response in the plant is required as a reaction to this alarm, this response must be programmed in CFC via the Disturbance output [Dstb].

No feedback monitoring

If no feedback monitoring is required, and the address string is left blank, the monitoring periods are used by the block for the internal generation of the transient state [TraSta]. This means that the transient state signal for the switch-on/off operation is set for the preset period of time. This is how a moving actuator, e.g., a damper, is displayed in the system.



Limit monitoring

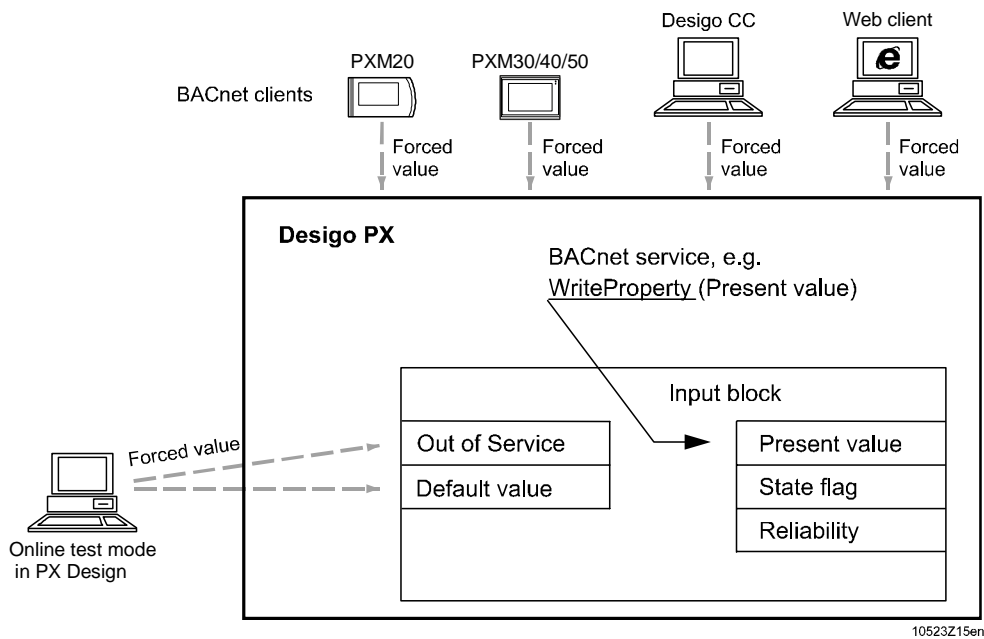
Blocks: AI, AO, AVAL

In the case of the analog I/O blocks, the present value [PrVal] can be monitored for a high/low limit. If the alarm monitoring feature is enabled, a deviation alarm is generated after a configurable time period, and the block status changes to *In Alarm*. When the present value is within the limits again and the configured time period has expired, the alarm and status are reset. There is otherwise no automatic block reaction, that is, if a switch response in the plant is required as a reaction to this alarm, this response must be programmed in Xworks Plus (XWP) via the disturbance output [Dstb].

Override via client

The input, output and value blocks can be overridden via BACnet clients or in XWP (CFC) in online test mode.

User override of an input value



10523Z15en

There are two options:

1. Override via a BACnet client:

A BACnet client is overridden with a BACnet service.

Input objects are overridden by setting the out-of-service parameter [OoServ] and writing the desired present value [PrVal]. The default value [DefVal] is automatically set to the same value as [PrVal]. (You can also overwrite [DefVal], in which case [PrVal] is automatically used instead).

There is no need to follow these rules when using the PXM20 to override a value, as the operator unit observes them automatically.

Overridden input objects are not reset automatically. To do this, reset [OoServ] first. [DefVal] remains at the last overridden value and [PrVal] is again derived from the physical input.

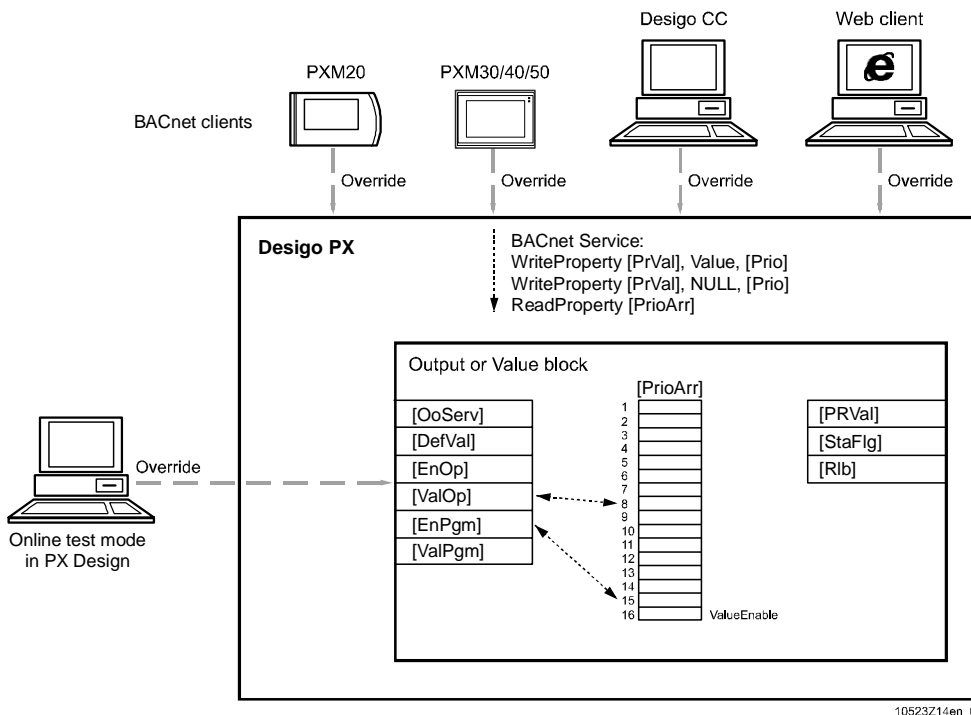
2. Override via online test mode in CFC:

Overrides with CFC are carried out via a proprietary service.

Outputs of a block cannot be overwritten.

To overwrite [PrVal] the out of service state in [OoServ] must be set to TRUE, after which the default value [DefVal] can be modified. This value is then adopted (or applied) as the present value and is made available under [PrVal].

User override of an output value



There are two options:

1. Override via a BACnet client:

The override of an output or value object is based on the priority array [PrioArr] in the object. Priority 8 is reserved for the operator, that is, an override from the PXM20 and Web client is written to the Priority 8 entry. Other BACnet clients can write to other priority entries.

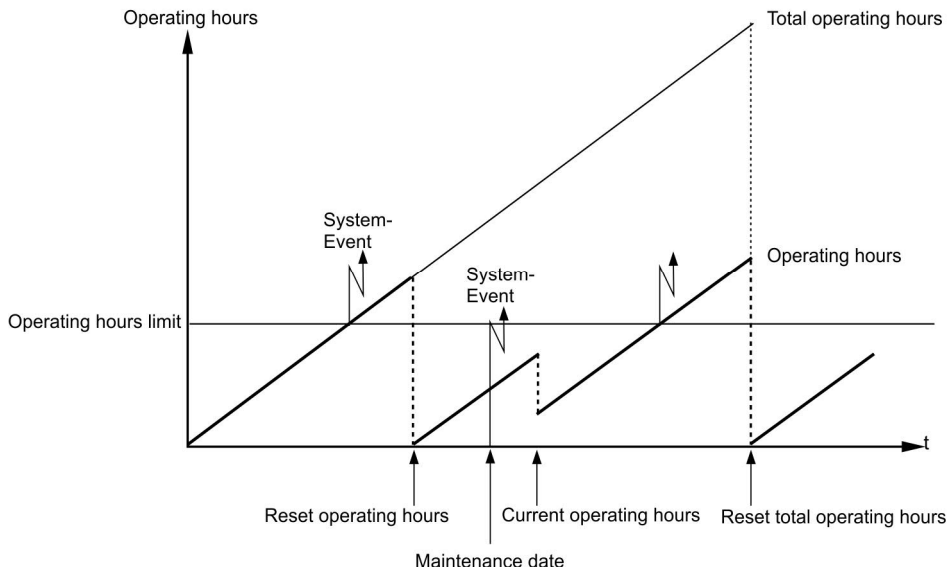
The value (Value or Null) is stored in the [PrioArr]. After processing in the object, the value, other than NULL, with the highest priority is transmitted to [PrVal]. If there is no active priority, the [DefVal] is transmitted.

2. Override via the online test mode in Xworks Plus (XWP):

[PrVal] is an output and can therefore not be modified. In this case the value under [EnOp] must first be set, after which the modifiable value under [ValOp] is written to Priority 8 in [PrioArr]. After processing in the object, the value, other than NULL, with the highest priority is then transmitted to [PrVal]. If there is no active priority, the [DefVal] is transmitted.

Runtime totals

Runtime totalization can be implemented in the binary input, binary output and multistate input and output blocks (BI, BO, MI and MO). Part of the overall range of functions is defined by the BACnet standard. In order to provide the complete range of runtime totalizing functions required in the field of building automation and control, certain proprietary enhancements have been added here.



Function

With a binary input object, the operating hours are determined on the basis of the ON state of [PrVal] (that is, by measuring the time for which this value is active). For multistate blocks, you can configure how many states are to be totalized. These are combined and added in a totalizer (the various states cannot be evaluated individually). In contrast to the input object, the output objects of the ON state for [FbVal] is logged (not [PrVal]) to operating hours message of the output objects.

There are two separate totalizers for runtime totalization:

- Runtime totalizer
- Overall runtime totalizer

Release

Runtime totalization can be enabled via the [EnOph] pin (Enable operating hours count). This is a binary value for binary objects, for multistate objects a list of values released for counting.

Runtime totalizer

Maintenance messages (events) are generated via the runtime totalizers. These are typically reset when the maintenance has been carried out. The present operating hours [PrOph] output can be used to connect the runtime totalization feature for further use in the program (e.g., for changeover of pumps or boilers based on operating hours).

Resetting the runtime total

The operating hours [Oph] input is used to reset the current runtime total. In online test mode in Xworks Plus (XWP) via a BACnet client the present value can be reset by overwriting it with a new value (usually 0). This reset does not affect the total operating hours count (pins [OphTot] and [PrOphTot], total operating hours and present total operating hours respectively).

Overall runtime totalizer

The total operating hours count records the total hours run by an aggregate. It is only reset when the aggregate is replaced. The [PrOphTot] output is available for further interconnection in the program.

Resetting the operating hours total

The [OphTot] input is used to reset the total operating hours. In online test mode in Xworks Plus (XWP) or via a BACnet client, the present value can be reset by overwriting it with a new value (usually 0). This reset procedure simultaneously sets the runtime totalizer (pins [Oph] and [PrOph]) to the same value.

This is necessary, e.g., for an aggregate which is installed as a replacement item, but which has previously been in operation elsewhere for some time.

Maintenance message

A maintenance message (event) can be generated either after a specified period of operation or on a specified date. The operating hours limit value and the maintenance date [OphLm]/[MntnDate] can be configured for this purpose. An event message is generated when the limit value is exceeded or at 13:00 hours on the preset date. At the same time, the binary output [MntnInd] (maintenance indication) is set to active for further use in the program. After the operating hours reset, this output reverts to inactive. At the same time, the time stamp of the last reset is stored in the time stamp operating hours reset pin [TiStmOph].

Feedback value

The following applies to output blocks: When a feedback is configured, operating hours count is done based on the feedback value and not based on present value.

The maintenance interval can be further connected via the output present total operating hours limit [PrOphLm].

Value range for run time totalizing

The hours run are registered in 32-bit format, giving a maximum value of 4,294,967,296. With a resolution in seconds, this gives a value range of over 49,000 days (more than 136 years).

Out of Service [OoSrv]

The physical input/output is disconnected from the I/O block via the out-of-service pin [OoSrv]. This out of service function is normally used in cases where a hardware module is faulty or temporarily not required, e.g., sensor not connected or faulty. This is a way of suppressing reliability problems and the associated FAULT alarms.

Input block

If the out-of-service property of an input block is set [OoSrv=True], the physical input is disconnected from the present value ([PrVal] = [DefVal]) and any changes in the physical input will not be transmitted to [PrVal]. Furthermore the reliability [Rlb] and status flag [StaFlg] are also disconnected from the physical input. In this state, the properties [PrVal] and [Rlb] can be modified for test purposes.

Output block

If the out-of-service property for an output block is set [OoSrv=TRUE], the physical output is disconnected from [PrVal]. Changes in [PrVal] will not be transmitted to the physical output, which retains its last value. Furthermore, the reliability [Rlb] and status flag [StaFlg] are also disconnected from the physical output. In this state, [PrVal] and [Rlb] can be modified for test purposes. Other functions that depend on these properties are not dependent on the [OoSrv] property. The [PrVal] is set in accordance with the priority array [PrioArr], but the value is not transmitted to the physical output.

Alarm and event functions

Each input, output and value block can be enabled and disabled as an alarm source. The blocks are configured by setting the relevant values at the block pins. See Alarm Strategy.

Reliability [Rlb]

The reliability of the present value and of the physical input/output is represented by the reliability pin [Rlb]. This makes it possible to detect and signal faults and errors, such as addressing errors, sensor problems (short-circuit or open circuit) and module faults (missing or incorrect modules). See Reliability Table.

Commissioning State [ComgSta]

The state of the I/O can be entered at [ComgSta], the commissioning state pin, in the commissioning phase. The setting does not affect the program; it merely serves as a kind of notepad for commissioning purposes.

The following states are available for selection:

- Checked
- Not Checked [DefVal]
- Periphery Defect or Missing
- Cable Defect or Missing
- I/O Defect or Missing

As these states are static, they must be set manually during commissioning.

Status Flag [StaFlg]

The status flag [StaFlg] indicates the state of the I/O block. This pin consists of four Boolean values:

- IN_ALARM: Logic 1 (TRUE) if the event state pin [EvtSta] does not display NORMAL as its value.
- FAULT: Logic 1 (TRUE) if the [Rlb] pin does NOT display the value NO_FAULT_DETECTED.
- OVERRIDDEN: Logic 1 (TRUE) if the block point was overridden locally (e.g., manual switch on I/O module). If this flag is set, [PrVal] and [Rlb] will no longer display any changes in the physical input/output.
- OUT_OF_SERVICE: Logic 1 (TRUE) if the out-of-service pin [OoSrv] is active.

Default Value [DefVal]

For an input block, [DefVal] is transmitted to [PrVal] when [OoServ] is set to TRUE.

For an output block, value [DefVal] is transmitted to [PrVal] when none of the priorities (1...16) is active.

19.2 Input blocks

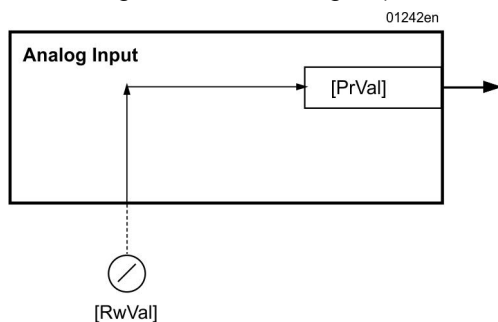
An input block is used to enable an input signal (e.g., a measured value) in the program to be handled as a process value.

Analog Input (AI)

The analog input is the logical image, or memory map, of an analog measured value and describes its properties. The raw data is converted and made available in the form of a current value (Present Value) at the block output for further processing within the program.

The following functions are integrated in the block:

- Conversion of the input signal with slope [Slpe] and intercept [lcpt].
- Interruption of input signal [OoServ] and replacement with [DefVal]
- Limit value monitoring (OFFNORMAL alarm)
- Reliability monitoring [Rlb] (FAULT alarm)
- Change of state messages (events / system events)



Processing and displaying the current value

The measured raw value is converted into the measured present value in accordance with a conversion curve. This present value is available at [PrVal] for further processing within the program.

Slope/Intercept

The conversion curve is a linear function which takes the following form:

$$[\text{PrVal}] = \text{Raw value} * \text{Slope} + \text{Intercept}$$

The values for slope [Slpe] and intercept [lcpt] must be defined specifically for the application concerned in accordance with the I/O system in use and the signal type.

For slope and intercept values for SBT products, see *Slope [Slpe] and Intercept [lcpt]*. For sensors not listed, the following applies:

Calculating [Slpe] and [lcpt]

The values for [Slpe] and [lcpt], which are to be entered in the block, must first be calculated. These values are derived from the individual [Slpe] and [lcpt] values of the signal type and the signal transducer in accordance with the following formula:

$$[\text{Slpe}] = (\text{Slope}_{\text{SignalType}} / \text{Slope}_{\text{SignalTransducer}})$$

$$[\text{lcpt}] = (\text{Intercept}_{\text{SignalTransducer}} / \text{Slope}_{\text{SignalTransducer}}) + \text{Intercept}_{\text{SignalType}}$$

[Slpe] is calculated on the basis of:

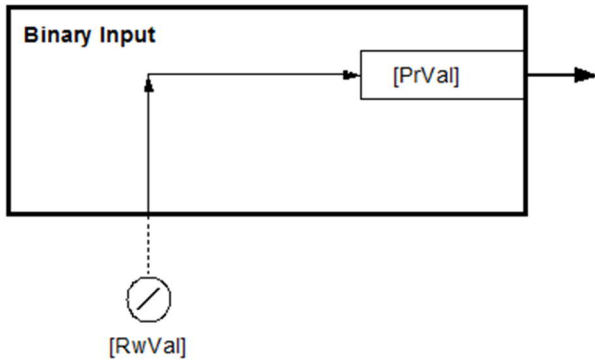
$$[\text{Slp}] = (\text{InterpolationPoint}_y2 - \text{InterpolationPoint}_y1) / (\text{InterpolationPoint}_x2 - \text{InterpolationPoint}_x1)$$

Binary Input (BI)

The binary input block is the logical image, or memory map, of a binary switch value and describes its properties. The parameters of the physical value are set via the polarity [Pol], and the value is then available as the present value for further processing. The Present Value is monitored for a given state. For commissioning and test purposes, or in the event of an error, the Present Value can be dissociated from the process and overwritten with a replacement value.

The following functions are integrated in the block:

- Inversion of the input value
- Interruption of input signal [OoServ] and replacement with [DefVal]
- Alarm value monitoring (OFFNORMAL alarm)
- Reliability monitoring [Rlb] (FAULT alarm)
- Change of state messages (events / system events)
- Runtime totalization and maintenance messages

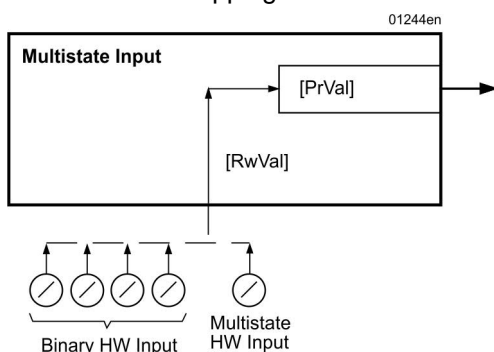


Multistate Input (MI)

The multistate input block is the logical image, or memory map, of several binary switch values or a direct hardware multistate value, and describes its properties. The multistate capability is achieved by interconnecting a number of individual binary states. The binary states are evaluated and mapped as integers. Each integer in the series is allocated a text label which is further processed and interconnected within the program as a current value. For commissioning and test purposes, or in the event of an error, the Present Value can be dissociated from the process and overwritten with a replacement value. As an auxiliary function, the runtime total for this multistate input can be registered and evaluated.

The following functions are integrated in the block:

- Interruption of input signal [OoServ] and replacement with [DefVal]
- Alarm value monitoring (OFFNORMAL alarm)
- Reliability monitoring [Rlb] (FAULT alarm)
- Change of state messages (events / system events)
- Runtime totalization and maintenance messages
- Hardware mapping



Pulse converter (pulse counter)

The pulse converter object cumulates pulses for a meter. The Pulse converter object is used where meter values already manipulate in a meter object or where changes of values are required to further process control programs. Applications include: Establishing 24-hour/7-day/monthly meters, transmission by the minute of meter values to peak load programs, etc. Precision and round off error based on real arithmetic is possible.

Specific properties

The counter value is scaled as a REAL number directly in the object using the scaling factor. COV forming the Present_Value can be value or time-related and a timestamp with the logged time is always provided with the Present_Value. Reduction of Present_Value by a value (subtraction) is supported as a standard. You can set it to a pre-defined value using a trigger function (proprietary expansion).

The Pulse Converter object can be used in two different manners: Counting or metering. The type of application is parameterized using the FncMod parameter.

The referenced object, e.g., an external device provides the pulse value:

- Present_Value for the pulse converter object represent the pulse count of the referenced object: The difference to the last read value is added for each record.
- Present_Value can be set via the system.
- After start-up, the pulse converter object encompasses the last stored counter value:
- After a change in counter, the pulse converter object encompasses a false counter value.
- Typical application: On-board I/O with pulse logging.

The referenced object, e.g., an external device provides the absolute pulse value:

- Present_Value from the pulse converter object represents the absolute counter value of the referenced object.
- Under no circumstance may the Present_Value be set via the system.
- After start-up or a change in counter, the pulse converter object after includes the correct counter value.
- Typical applications:
 - Access to an accumulator or pulse converter object is another BACnet device
 - I/O Open module or M-bus with counter value integration
 - Integration of a device via LON
- Incorrect applications: I/O module with pulse recording

Accumulator object (counter value)

The accumulator object can map counter states unchanged and free of errors due to rounding off or add the counter pulse without loss and scale the same. The accumulator object is suitable to displaying meter values that justify monetary performance. For this type of counter values, manipulations such as monthly values, etc., must never be made directly in the meter object.

The addition of counter pulses and scaling without loss is accomplished using whole-number operations with residual value processing. The conversion of physical pulses can be adapted using a presale parameter. The resulting Present_Value is a scalable variable.

Present_Value depends on the function mode to synchronized adjustable to any value using a physical meter with the last value prior to setting saved with a date/time stamp.

19.3 Output blocks

An output block is the logical image or memory map of a command, and describes its properties. Within the program, the Present Value is made available to the block as a program value. The block converts the program value and transmits the raw data to the physical I/O.

If an output is deleted from an existing system in the course of a modification, the I/O module will retain the last valid value which it received from the system. You can return the I/O channel to the default status by switching the power off and on again. This problem can be avoided by performing a complete download.

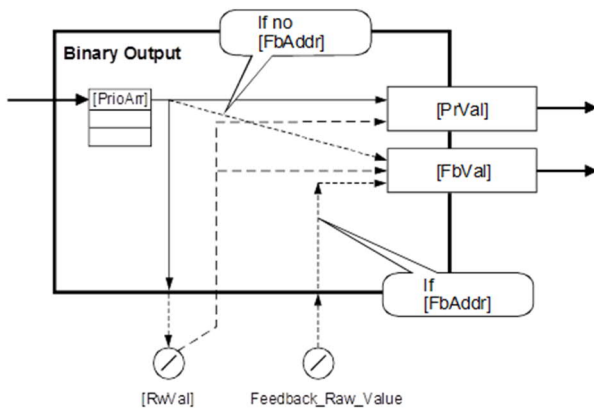
Binary Output (BO)

The binary output block is the logical image, or memory map, of a binary switch command and describes its properties. Within the program it is made available to the block as a program value, and its parameters are set via the "Polarity" pin. The block converts this program value and transfers the raw data to the physical I/O, where it is converted into a digital signal, e.g., which drives the field device via a contact.

The following functions are integrated in the block:

- Evaluation of the priority array [PrioArr]
- Inversion of the switch value and the feedback value (Polarity of feedback [Bop])
- Interruption of the output signal [OoSrv]
- Feedback monitoring (OFFNORMAL alarm)
- Reliability monitoring [Rlb] (FAULT alarm)
- Change of state messages (events / system events)
- Configurable switch types (Normal, Trigger, Pushbutton, Pushbutton with delay)
- Runtimes and monitoring periods

- Switch-command delays
- Process monitoring [StaFlg]
- Runtime totalization and maintenance messages



Feedback monitoring for dampers with one end switch

To monitor the damper position of dampers with one end switch, the switch position must be set by defining the polarity of the feedback signal [Bop].

OPEN end switch -> Feedback polarity [FbPol] set to NORMAL

CLOSED end switch -> Feedback polarity [FbPol] set to INVERTED

Feedback monitoring for dampers with two end switches

The monitoring of dampers with two feedback signals (Open/Closed) is implemented via the address string of the Feedback Address [FbAddr]. The first address in the string must be that of the end switch which indicates that the damper is closed. The end switch indicating that the damper is open is set in the second part of the address string.

Example with PX modular:

P= M1.K1; M2.K2 (D20)

- 1. 1st address: Damper-CLOSED switch
- 2. 2nd address: Damper-OPEN switch
- Feedback polarity [FbPol] NORMAL

M1.K1 = True; M2.K2 = False -> Feedback value: Closed

M1.K1 = False; M2.K2 = True -> Feedback value: Open

When the damper is being driven to the OPEN or CLOSED position, this transient state [TraSta] is displayed. If the preset monitoring time is exceeded, an alarm is initiated. If the damper fails to reach an end position, the alarm is reset again after the monitoring time has expired. There is otherwise no automatic block reaction, that is, if a switch response in the plant is required as a reaction to this alarm, this response must be programmed in CFC via the disturbance output [Dstb].

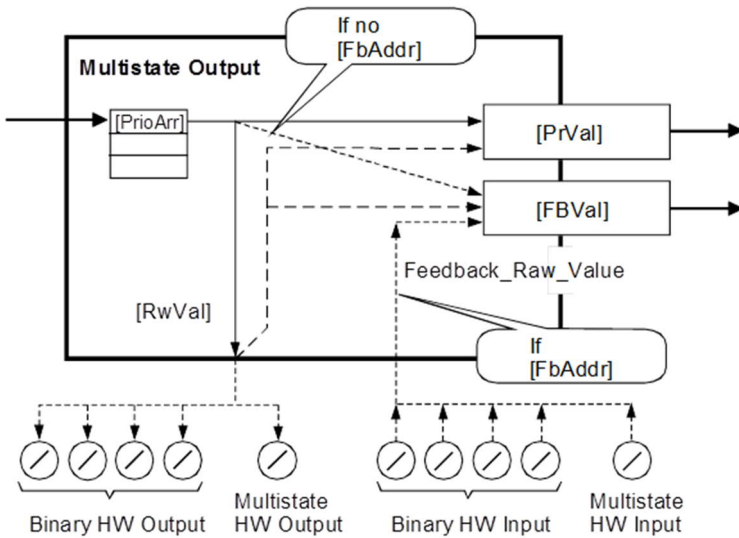
Multistate Output (MO)

The multistate output is the logical memory map of a multi-state switching command, and describes its properties. Within the program, the current value is made available as a program value to the block and transmitted after conversion into raw-data format to the physical I/Os. Here the raw data is converted into a digital signal, e.g., which drives the field device via a contact. It is also possible to connect a multistate feedback signal, which is used for alarm evaluation.

The following functions are integrated in the block:

- Evaluation of the priority array [PrioArr]
- Interruption of the output signal [OoServ]
- Feedback monitoring (OFFNORMAL alarm)
- Reliability monitoring [Rlb] (FAULT alarm)
- Change of state messages (events / system events)
- Configurable switch type (Normal, Motor, Trigger)
- Runtimes and monitoring periods
- Hardware mapping (refer to Section 0)

- Runtime totalization and maintenance messages
- Process monitoring [StaFlg]

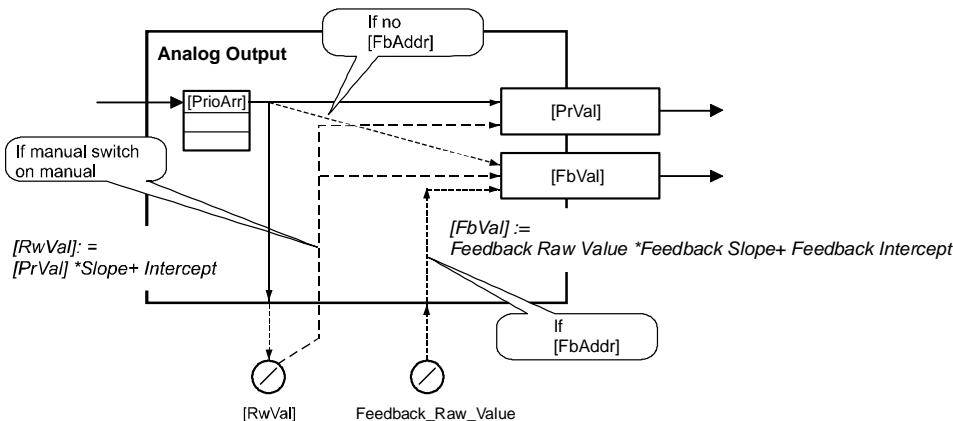


Analog Output (AO)

The analog output is the logical image, or memory map, of an analog control command and describes its properties. Within the program, the Present Value is made available to the block as a program value. The block converts the program value and transfers the raw data to the physical I/O, where it is converted into a 0...10 V signal, e.g., to drive a field device.

The following functions are integrated in the block:

- Evaluation of the priority array [PrioArr]
- Interruption of the output signal [OoSrv]
- Conversion of the process value and feedback signal with slope [Slpe] and intercept [Icpt]
- Configurable switch type (Normal or Trigger)
- Limit value monitoring (OFFNORMAL alarm)
- Deviation monitoring
- Reliability monitoring [Rlb] (FAULT alarm)
- Change of state messages (events / system events)
- Process monitoring [StaFlg]



The value [PrVal] from the program is converted into the physical positioning value by use of a conversion curve. This present value is then available at [PrVal] for further processing in the program while at the same time, the raw data is transmitted to the associated I/O system, where it is converted into an electrical signal to drive the field device.

The conversion curve is a linear function which takes the following form:

$$\text{Raw Value [RwVal]} = [\text{PrVal}] * \text{Slope} + \text{Intercept}$$

The values for slope [Slpe] and intercept [Icpt] must be defined specifically for the application concerned in accordance with the I/O system in use and the signal type.

For slope [Slpe] and intercept [Icpt] values for SBT products, see *Slope [Slpe] and Intercept [Icpt]*.

19.4 Value objects

Value objects can be seen as virtual data points which are defined in the BACnet standard and have the same functions as the I/O blocks.

- Analog value block
- Binary value block
- Multistate value block

The only difference, in the case of value blocks, is that it is not possible to define physical connections to sub-components or components (e.g., to I/O modules) in the plant. The value objects BVAL, AVAL and MVAL are used in the program whenever BACnet-defined functions, such as commands, alarm generation and runtime totalizing are required, or when a value is to be modified via an operator unit. Value blocks look like all other blocks, and can be connected with other blocks.

Typical applications

Value objects are used typically in aggregates as command control links (PWR_CTL or CMD_CTL). The command control mechanism passes the commands to the value object and derives the status from the BACnet referencing system.

Higher level plant control
Hierarchy level: Partial plant
- Plant enable
- Scheduler
- Plant switch...

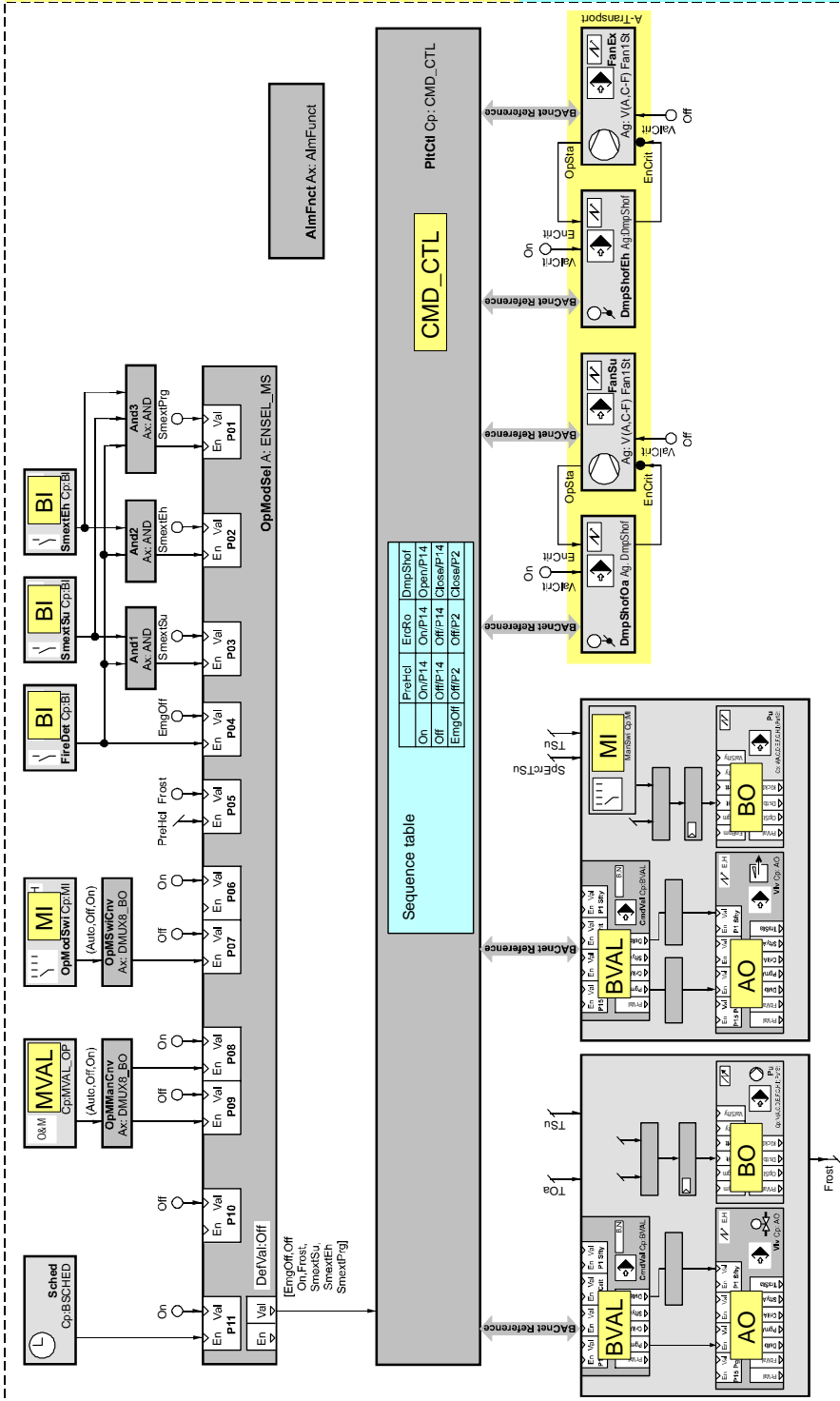
- Response to faults

- Sequence control

- Switch on
- Switch off
- Failsafe function
- Smoke
- Frost
- Night cooling
- Purge

Local control
Hierarchy level: Aggregate
- Local response to faults
- Interlocks
- Failsafe values
- Local operat. intervention

103983707en



Furthermore, the value objects can be used for alarm monitoring (reference values or high/low limit value), or to determine and monitor operating hours. The value objects designed specially for operation via BACnet client can be used, e.g., as a simple way of enabling the user to operate setpoints and switch commands.

Analog Value (AVAL)

The analog value block provides access to the dataflow, that is, to signals and pins with a Real number as the data type. The value objects can be inserted into the program structure and interconnected in any configuration.

This block is used when, e.g.:

- An alarm is to be created within the CFC chart as a commandable interface of an aggregate (e.g., limit monitoring of an output value of an aggregate).
- Access from the operator unit is required, in order to modify a value

The following functions are integrated into the block:

- Evaluation of the priority array [PrioArr]
- Interruption of the output signal [OoServ]
- Limit value monitoring (OFFNORMAL alarm)
- Deviation monitoring
- Reliability monitoring [Rlb] (FAULT alarm)
- Change of state messages (events / system events)
- Process monitoring [StaFlg]

Binary Value (BVAL)

The binary value block provides access to the dataflow, that is, to signals and pins with a Boolean number as the data type. The value objects can be inserted into the program structure and interconnected in any configuration.

This block is used when, e.g.:

- An alarm is to be generated as the commandable interface of an aggregate (e.g., monitoring of logic operations)
- The hours run are to be totalized after a logic operation
- Access from the operator unit is required, in order to modify a value
- Configurable switch types (normal, switch, pushbutton with delay)

The following functions are integrated into the block:

- Evaluation of the priority array [PrioArr]
- Interruption of the output signal [OoServ]
- Process value monitoring (OFFNORMAL alarm)
- Reliability monitoring [Rlb] (FAULT alarm)
- Change of state messages (events / system events)
- Configurable switch types (normal, switch, pushbutton with delay)
- Runtimes and monitoring periods
- Switch-command delays
- Process monitoring [StaFlg]
- Runtime totalization and maintenance messages

Multistate Value (MVAL)

The multistate value block provides access to the dataflow, that is, to signals and pins with a Multistate number as the data type. The value objects can be inserted into the program structure and interconnected in any configuration.

This block is used when, e.g.:

- An alarm is to be generated as the commandable interface of an aggregate (e.g., for limit monitoring)
- Access from the operator unit is required, in order to modify a value
- Hours run are to be totalized

The following functions are integrated in the block:

- Evaluation of the priority array [PrioArr]
- Interruption of the output signal [OoServ]
- Process value monitoring (OFFNORMAL alarm)
- Reliability monitoring [Rlb] (FAULT alarm)

- Change of state messages (events / system events)
- Runtimes and monitoring periods
- Runtime totalization and maintenance messages
- Process monitoring [StaFlg]

19.5 Value objects for operation

To simplify operation, use the value objects BVAL_OP, AVAL_OP and MVAL_OP. The blocks are specifically intended for the operation of setpoints via BACnet clients. They do not require a manual override from the operator unit. Value objects look like all other blocks, and can be connected with other blocks. The blocks do not include alarm generation or runtime totalization.

19.6 Addressing the I/O blocks

Hardware independence

Logical I/O blocks allow the standardization of the inputs and outputs irrespective of the hardware. The relationship between a given logical I/O and its physical equivalent is established by assigning the address of the I/O system concerned.

This independence has the advantage that the functions of the block, as defined by the BACnet standard and the specific Desigo PX enhancements, do not change. The number of different I/O systems or physical I/Os can be expanded freely.

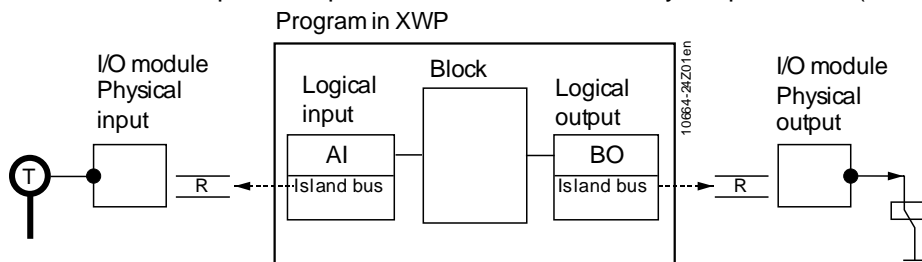
Identical compound libraries

Another advantage is that the compound libraries are always identical. In the engineering phase, they are adapted to the I/Os in the project by assigning the appropriate addresses. The process values (0...10V, 0...25mA, signal contacts, etc.) from the connected field devices are registered directly at the physical inputs. The physical outputs deliver the process values (0...10V, switching stages 0 / I / II / III, etc.) directly to the connected field devices.

The process values are transmitted in the form of raw data via the relevant medium (e.g., PPS2); the conversion of the raw value takes place within the block.

Rules:

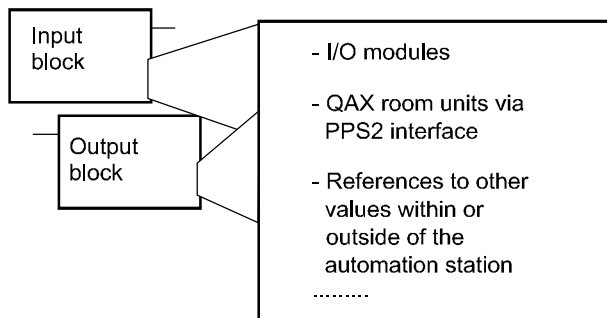
- Values from the plant are measured and processed in Input blocks (Analog, Binary or Multistate).
- Values to the plant are processed and transmitted by Output blocks (Analog, Binary or Multistate).



I/O systems

To enable the process value of the logical I/O block to be allocated to the appropriate physical I/O, the relevant address must be assigned. The address is assigned as follows:

- Via automated assignment by the Point Configurator to the CFC
- Direct allocation to the I/O block in Xworks Plus (XWP)



The logical I/O blocks are designed for universal use in various I/O systems. The specific address structures and hardware definitions are determined by the I/O system, e.g., the failsafe control value for the island bus. In Desigo, they are as follows:

- Physical I/Os
- Values in a Desigo room unit, accessible via the PPS2 interface
- Data in the same or in another automation station, referenced by its Technical Designation and accessed without a connection, peer-to-peer via BACnet services.

Address prefix

The addressing syntax indicates the origin of the raw value. The syntax must correlate with the actual physical inputs.

The prefixes for the various subsystems are as follows:

- "T=" for TX-I/O modules on an island bus-capable automation station PXC...D
- "C=" for on-board I/Os of the Desigo PX compact automation stations
- "B=" for referencing to BACnet objects
- "Q=" for QAX room units
- "L=" for LonWorks addressing
- "M=" for PX-OPEN addressing
- "D=" for PX Open Diagnostic Addressing

For addressing with "P=", see *Addressing Entries for PXC...-U, PTM and P-Bus*.

For addressing with "S=", "M=" and "D=", see the corresponding expert documentation.

For more information on TX-I/O, see *TX-I/O Assortment overview (CM2N8170)* and *TX-I/O Functions and operation (CM110561)*.

Addressing entries PX Modular (PXC100/200..D)

For PX compact, the TX-I/O module at the [IOAddr] pin start with a "T" (prefix: "T=").

Address syntax:

T= Module.I/O point (signal type)

Example: T=2.1 (Y10S)

The parameters no longer appear in the I/O address string for direct island bus integration, but rather in the IOC (I/O configuration).

The only exception is the Info LED which must have the prefix "C=", because the fixed address, 8.1, which is used for the Info LED may also be used by an I/O module.

The Info LED for PX KNX and PX Open can also be addressed with C=8.1.

Required address entries when using the modular series automation station in conjunction with TX-I/O-I/O modules

Signal types shown in *italics* are used to map virtual modules for use with TX OPEN at module level. Signal types AIS, AOS, DIS and DOS deliver a 16 bit value with status information, while signal types AISL, AOSL, DISL and DOSL deliver a 32 bit value with status information. All other signal types deliver a 16/32 bit value without status information.

While all the module types listed may be connected to any island bus addresses, not all module types have 16 points.

| Type | Module addressing | I/O point |
|---------------|-------------------|-----------|
| Desigo TX-I/O | 1...120 | 1...16 |
| PX Info LED | 8 | 1 |

| Module type | Signal type | Example |
|--------------|--|-------------|
| Analog Input | R1K, P1K, P100, U10, I25, I420 R2500, R250 (only TX-I/O) T1, NTC10K, NTC100K (only TX-I/O) | T=1.1 (R1K) |
| | <i>AI, AIS, AIL, AISL</i> | T=2.1 (AIS) |

| Module type | Signal type | Example |
|-------------------|--------------------------------------|---|
| Analog Output | Y10S | T=2.1 (Y10S) |
| | Y250T PWM | T=3.1 (Y250T) |
| | Y420 AO, AOS, AOSL, AOL | T=34.1 (Y420) T=36.1 (AOS) |
| Binary Input | D20, D20S D42, D250 (only PT-I/O) | T=25.2 (D20) |
| | DI, DIS, DIL, DISL | T=26.3 (DIS) |
| Counter Input | C | T=38.1 (C) |
| Info LED | Q_LED | C=8.1(Q_LED) |
| Binary Output | Q250_P, Q250A_P | T=12.1 (Q250_P) |
| | Q250 QD, Q250B, (only PT-I/O) | T=1.1 (250) T=14.1 (Q250) + T =15.1(D20) |
| | DO, DOS, DOL, DOSL | T=15.2 (DOS) |
| Multistate Input | D20 D42, D250 (only PT-I/O) | T=1.1 (D20) + T=1.2 (D20) -- |
| | DI, DIS, DIL, DISL | T=7.1 (DIS) |
| | | |
| Multistate Output | Q250-P1 ... Q-P5 | T=1.1 (Q250-P3) |
| | Q-M1 ... Q-M4 QD-M2 (only PT-I/O) | T=1.1 (Q-M3) -- |
| | | |
| | DO, DOS, DOL, DOSL | T=26.3 (DIS) |

Parameter values

Parameters are entered in the I/O address editor.

See *Automation stations modular series PXC..D, PXC..E.D, PXA40..* (CM1N9222).

Addressing entries PX Compact (PXC...)

The addressing procedure is almost identical for Desigo PX compact and for Desigo PX modular. However, the valid address ranges and signal types are not the same as those used for the addressing of individual TX-I/O modules.

For PX compact, the "on-board" I/O modules at the [IOAddr] pin start with a "C" (prefix: "C="). Address syntax:

C=Module.Channel (signal type, parameter)

Example: C=2.1 (Y10S, NO)

The table below shows the available address ranges and signal types, which vary according to the Desigo PX compact automation station (each with its own integrated, fixed configuration of I/Os) type.

The existing UI and AO can be configured as AI, DI, CI, or AO.

Signal type of no application is loaded (wiring test):

PXC12..D, U1...U4: xx = Y10S, U5...U8: xx = R1K

PXC22..D, U1...U4: xx = Y10S, U5...U16: xx = R1K

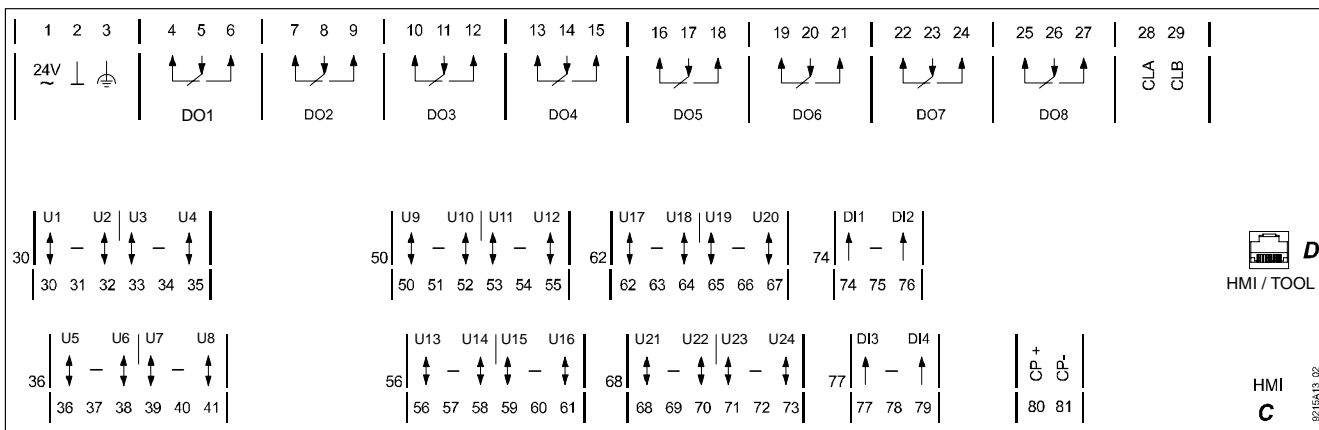
PXC36..D, U1...U6: xx = Y10S, U7...U24: xx = R1K

Addressing entries PX compact

| PX compact | | PXC12.D PXC12-E.D | | PXC22.D PXC22-E.D | | PXC36.D PXC36-E.D | |
|---------------|---|----------------------|---|----------------------|---|----------------------|--|
| | | Module | | Channel | | Module | |
| UIO | 1 | 1..4 | 1 | 1..12 | 1 | 1..18 | R1K, U10, T1, N1K, P1K, C, D20, D20S |
| Universal I/O | | U5..U8 | | U5..U16 | | U7..U24 | |

| PX compact | | PXC12.D PXC12-E.D | | PXC22.D PXC22-E.D | | PXC36.D PXC36-E.D | |
|-------------------------|---|----------------------|---|----------------------|---|----------------------|------------------------------|
| | | Module | | Channel | | Module | |
| UIO | 4 | 1..4 | 4 | 1..4 | 4 | 1..4 | R1K, U10, T1, |
| Universal I/O with Q250 | | U1..U4 | | U1..U4 | | U1..U6 | N1K, P1K, C, D20, D20S, Q250 |

Layout of PXC36.D housing with address ranges

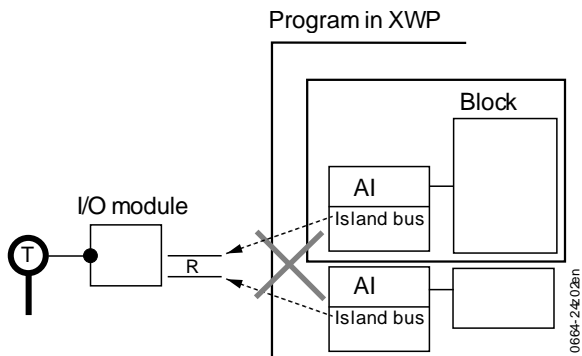


See *Automation stations, compact model PXC..D* (CM1N9215).

Multiple use of sensors

Multiple use of I/O signals

Multiple use by addressing the physical I/Os in two or more logical I/O blocks (as shown in the following figure) is not allowed.



If you wire it as in the figure above, Xworks Plus (XWP) determines multiple use and generates an error message.

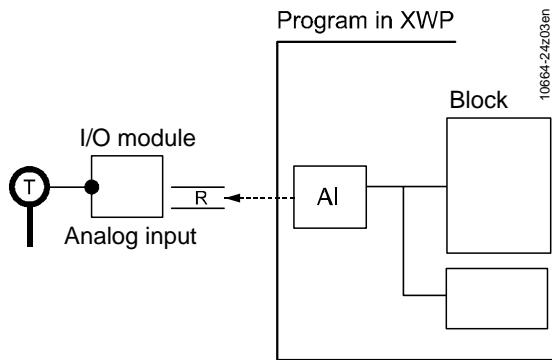
For the multiple use of output blocks, the plant will malfunction, because there will then be two or more sources acting on one switching command. The effective switching command (at the output) is the last one received (determined by the rule "the last command takes precedence"). In other words, the order of processing determines which source or origin will be linked to the output.

In CFC the same address can be allocated to two or more input or output blocks. This multiple address allocation goes undetected when the program is compiled; the automation station also fails to recognize the error (a reliability error is generated and an error message is transmitted only in the case of multiple address allocation with two different signal types).

Solution 1

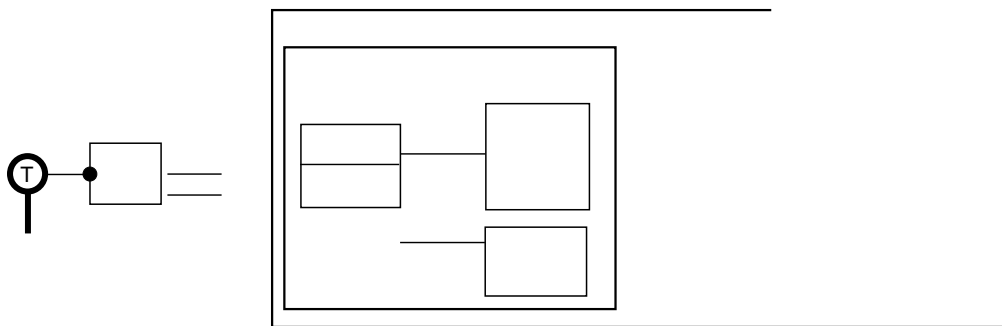
Many systems include a requirement for the multiple use of sensors. A typical example of this is an outdoor air temperature sensor shared across systems. The following example illustrates the simplest form of the multiple use of sensors:

In CFC the current value is transmitted for further use in the program by interconnecting the blocks. The logical I/O block (Analog Input, {AI}) occurs in the program once only, and its hardware-specific parameters only need to be set once.



Solution 2

The multiple use function can be implemented with a BACnet reference to the first analog input block (Partial plant 1). In other words, the first block will receive the island bus address at the [IOAddr] pin. The second analog input block (Partial plant 2) references the first AI (B=...) via the technical designation.



Addressing multistate I/Os

Multistate input

The multistate value is made up of several separate binary measured values.

Addressing is via the input/output address [IOAddr]. In both the modular and the compact series, the logical and physical I/O must be "located" in the same automation station, but they do not need to be contiguous (e.g., C=5.1;5.3;5.5;5.6(Q250) is valid). The addressing cannot extend across automation stations. The addresses must be on the same module for TX-I/O.

For information about addressing multistate I/Os with PTM, see *Addressing Multistate I/Os with PTM*.

Simple mapping

Syntax: T=Module.I/O point;Module.I/O point;Module.I/O point;Module.I/O point

Examples:

- T=1.1
- T=1.1;1.2
- T=1.1;1.2;1.3
- T=1.1;1.2;1.3;1.4
- T=10.3

Up to four binary status values (e.g., Off/St1/St2/St3/St4) can be registered. The signals to be registered, which are addressed via Module.Channel, must always be of the same hardware signal type. With the simple mapping procedure, to enable the multistate input to interpret the current binary signals correctly, only one binary signal may be present at any one time. If several binary signals are present at once, this is displayed as an error at the [Rlb] pin.

The examples below show a possible application for multistate input blocks in conjunction with the physical I/O modules. The example on the left of the diagram is a multiple I/O module, while the one on the right shows the mapping of several individual I/O modules in one multistate input block.

Multistate output

The multistate value from the program is converted in the Multistate Output block into a switching command. Addressing is via [IOAddr]. For PX modular, the syntax is as follows:

Syntax: T=Module.channel

Examples:

- Q-M1: T=1.1
- Q-M2: T=1.1
- Q-M3: T=1.1
- Q-M4: T=1.1
- Q250-P3: T=10.1
- DOS: T=24.7

Values with up to four stages can be processed. The signals to be registered, which are addressed via Module.Channel, must always be of the same hardware signal type. In the case of a multistate output on the hardware side, there is one address only (this is only possible with PXC modular automation stations).

Error handling

If an automation station does not support a given address (e.g., incorrect syntax) or a given I/O system, this will lead to a reliability error, which will be displayed at the [R1b] pin.

Advanced mapping (Multistate Input)

The manual switch can be encoded on the PX Compact in various ways, e.g.:

- (Auto/Off/On) or (Off/Auto/On)
- (Auto/Off/S1/S2) or (Off/Auto/S1/S2)

So avoid having to keep adapting the data types and text groups in the system, the manual switch must always be represented in the same way within the system:

- (Auto/Off/On)
- (Auto/Off/S1/S2)

A prerequisite for this approach is that it must be possible in the multistate input block to configure the hardware coding and mapping to the standardized manual switch. This is made possible with parameters in the address.

1_n-Mapping (Multistate Input and Output)

Syntax:

T = Module.channel

C=Module.channel;Module.channel;Module.channel;Module.channel (signal type, a,b,c,d,e)

a represents [PrVal] for HW-I/O (0,0,0,0)

b represents [PrVal] for HW-I/O (1,0,0,0)

c represents [PrVal] for HW-I/O (0,1,0,0)

d represents [PrVal] for HW-I/O (0,0,1,0)

e represents [PrVal] for HW-I/O (0,0,0,1)

Example: T=2.1

For the TX I/O addressing no additional information in the address string is added. All information (signal type, mapping table, mapping rules, e.g., up-down, etc.) is configured in the I/O Address Editor and loaded in the automation station with the IOC file.

Example: C=2.1;2.2;2.3;2.4 (D20, 2, 1, 3, 4, 5)

| [PrVal] | Addr1 | Addr2 | Addr3 | Addr4 | Comment / Text group |
|---------|-------|-------|-------|-------|----------------------|
| 2 | 0 | 0 | 0 | 0 | Off |
| 1 | 1 | 0 | 0 | 0 | Auto |
| 3 | 0 | 1 | 0 | 0 | Stage 1 |
| 4 | 0 | 0 | 1 | 0 | Stage 2 |
| 5 | 0 | 0 | 0 | 1 | Stage 3 |

Example: C=2.1;2.2;2.3;2.4 (D20, 2, 1, 5, 7, 9) ;-- with holes

| [PrVal] | Addr1 | Addr2 | Addr3 | Addr4 | Comment / Text group |
|---------|-------|-------|-------|-------|----------------------|
| 2 | 0 | 0 | 0 | 0 | On |
| 1 | 1 | 0 | 0 | 0 | Off |
| 5 | 0 | 1 | 0 | 0 | Comfort |
| 7 | 0 | 0 | 1 | 0 | Eco |
| 9 | 0 | 0 | 0 | 1 | StandBy |

UpDown Mapping (Multistate Input and Output)

Syntax:

Application: Connecting/disconnecting further stages.

Example: Electric heating registers, multi-stage burners.

T=Module I/O point

C=Module.channel;Module.channel;Module.channel;Module.channel (signal type, UPDOWN)

Example: T=2.1

For the TX I/O addressing no additional information in the address string is added. All information (signal type, mapping table, mapping rules, e.g., up-down, etc.) is configured in the I/O Address Editor and loaded in the automation station with the IOC file.

Example: C=5.1;5.2;5.3;5.4(Q250,UPDOWN)

Example: C=2.1;2.2;2.3;2.4(D20,UPDOWN)

| [PrVal] | Addr1 | Addr2 | Addr3 | Addr4 | Comment / Text group |
|---------|-------|-------|-------|-------|----------------------|
| 1 | 0 | 0 | 0 | 0 | Off |
| 2 | 1 | 0 | 0 | 0 | Stage 1 |
| 3 | 1 | 1 | 0 | 0 | Stage 2 |
| 4 | 1 | 1 | 1 | 0 | Stage 3 |
| 5 | 1 | 1 | 1 | 1 | Stage 4 |

With Up/Down mapping, more than one hardware input or output may be active.

Binary Mapping (Multistate Input and Output)

Application: Output of an integer in binary form.

Example: Binary electric heating coil.

Syntax: C=Module.channel;Module.channel;Module.channel;Module.channel (signal type, BINARY)

Example: C=5.1;5.2;5.3;5.4(Q250,BINARY)

Example: C=2.1;2.2;2.3;2.4(D20,BINARY)

| [PrVal] | Addr1 | Addr2 | Addr3 | Addr4 | Comment / Text group |
|---------|-------|-------|-------|-------|----------------------|
| 1 | 0 | 0 | 0 | 0 | Off |
| 2 | 1 | 0 | 0 | 0 | Stage 1 |
| 3 | 0 | 1 | 0 | 0 | Stage 2 |
| 4 | 1 | 1 | 0 | 0 | Stage 3 |
| 5 | 0 | 0 | 1 | 0 | Stage 4 |
| 6 | 1 | 0 | 1 | 0 | Stage 5 |
| ... | | | | | |
| 16 | 1 | 1 | 1 | 1 | Stage 15 |

With binary mapping, more than one hardware input or output may be active.

BACnet addressing

Peer-to-peer communication

Data can be exchanged via peer-to-peer communication.

The exchange takes place using the BACnet services defined in the BACnet standard. The process employs mechanisms engineered in CFC which can be tracked in online test mode, but which are based on BACnet objects and BACnet services.

Engineering

When engineering the exchange of data in CFC, it is important to take note of the following:

- Addressing is via [IOAddr].
- Data is exchanged only between BACnet objects. The attributes of the I/O blocks and pins must be defined appropriately, and the information must also be made available in the form of a BACnet object. For this purpose, the attributes of this block or I/O must be defined correctly.
- In BACnet terminology, the I/O block is a client which fetches the required value from an object defined as the server. This process is carried out using services defined by BACnet, e.g.: The client subscribes to the relevant object (the server) using the SubscribeCOV service. The server then supplies the value via the BACnet service COVReporting whenever it changes by the programmed value, COVIncrement. ReadProperty (polling) is another BACnet service. Here, the value is read at regular predefinable intervals.
- Addressing is carried out via the Technical Designation (TD). Note, however, that this Technical Designation must first be made known to the client in the form of a reference address.
- The data is exchanged both within a given automation stations, and across automation stations.

Address syntax

Addressing takes place via the input/output address [IOAddr] and always starts with the prefix "B=".

The BACnet reference address is the same as the Technical Designation (TD) of the value. The BACnet addressing syntax is as follows:

B=BACnetReference (BACnetConfig)

Example: B=Geb6'Lft3'FanSu'Mot'MntnSwi.PrVal(0)

Polling or COV procedure

The FB variable PollCyc is used instead of the prior BACnetConfig parameter in the I/O address syntax, to distinguish between COV or polling:

FB variable IOAddr. FB variable PollCyc

BACnetConfig = 0 -> COV (Change of Value)

BACnetConfig = 1...65535 -> Polling in seconds

In an automation station operating as a BACnet device, the maximum number of simultaneously supported COV subscriptions is limited to 400.

The BACnet Device as BACnet Server supports a maximum of 400 subscriptions from BACnet clients or from other BACnet devices via the BACnetReference.

A BACnet device operating as a BACnet client can also accommodate a maximum of 100 subscriptions to other values via the BACnetReference.

If the COV procedure is selected, COVIncrement is used for analog objects to define the value by which the present value must change to initiate a COV event.

Data output using WriteProperty

Output objects can write their Present_Value to the properties of other objects or command other value or output object.

Write without priority: Optional address string-Par(P=Number) no available.

Command with priority: Optional address string-Par(P=Number) available.

COV across sites

The value subscribed to must be available in the same BACnet network. Avoid a COV across sites.

The DeviceID is used to access and subscribe freely to values in different BACnet devices (especially in the case of third-party integration). The syntax is as follows:

B=[DeviceID]Objectname – where the object name can be any string required. The DeviceID is entered in decimal (instance number or entire ObjectID).

PPS2 addressing

A PPS2 address is required when values are to be transmitted via the PPS2 interface. Addressing takes place via the input/output address [IOAddr] and always starts with the prefix "Q=".

Address syntax

Up to five room units can be connected to one Designi PX automation station and addressed via the PPS2 interface. The addressing syntax is as follows:

Q=RoomUnitNumber.Object(Profile)

Example: Q=1.40 (1)

The functions available in the room unit are mapped directly to the I/O blocks. The following elements of the address are predefined:

| Type (standard BACnet objects) | Room unit number | Object | Object description | Profile ¹ | Example |
|--------------------------------|------------------|--------|--------------------------|----------------------|---------------------|
| Analog input | 1...5 | 24 | Setpoint correction | – | Q=1.24 |
| Analog output | 1...5 | 24 | Setpoint correction | – | Q=2.24 |
| Analog input | 1...5 | 40 | Room temperature | 0, 1...6 | Q=1.40 ¹ |
| Analog output | 1...5 | 195 | Room temperature display | – | Q=5.195 |
| Multistate input | 1...5 | 205 | Mode | – | Q=4.205 |
| Multistate output | 1...5 | 205 | Mode | – | Q=2.205 |
| Multistate output | 1...5 | 206 | Heating/Cooling display | – | Q=3.206 |

Key

¹ The Profile relates to the configuration number shown in the next table.

The room unit is configured with this configuration number and appended to the Room temperature object. Other objects are not assigned a configuration number. Only the relevant operating and process values are mapped in the I/O blocks, rather than all objects of a room unit.

Six profiles have been defined to keep both the memory requirements and the demands placed upon the user in practice to a reasonable level. If no profile information is supplied, the predefined device-specific default value [DefVal] is used. As an exception in the case of the QAX units, Profile No. 5 is used.

| Configuration | Profile | | | | | |
|------------------------------|---------|-----|-----|-----|----|----|
| | 1 | 2 | 3 | 4 | 5 | 6 |
| Enable operating mode | | | | | | |
| StandBy | ON | ON | ON | ON | ON | ON |
| Auto | ON | ON | ON | ON | ON | ON |
| Fan1 | ON | ON | ON | ON | ON | ON |
| Fan2 | OFF | OFF | ON | ON | ON | ON |
| Fan3 | OFF | OFF | OFF | OFF | ON | ON |
| KonfLCD | | | | | | |
| Symbol Standby | ON | ON | ON | ON | ON | ON |
| Symbol Auto | ON | ON | ON | ON | ON | ON |
| Symbol Fan1 | ON | ON | ON | ON | ON | ON |
| Symbol Fan2 | OFF | OFF | ON | ON | ON | ON |
| Symbol Fan3 | OFF | OFF | OFF | OFF | ON | ON |
| TempUnit | °C | °F | °C | °F | °C | °F |

This profile (or configuration number) is always valid for one room unit only. It is used to configure the objects ConfigLCD and EnableOperatingMode and to define how the room unit is to operate (e.g., °C or °F).

In principle, the profile can be attached to any other object.

This configuration applies only to the QAX33.1 and QAX34.1 room units.

Configuration of the object ConfigLCD is only relevant in the case of the QAX34.1, as this is the only unit with a display in °C or °F.

The configuration of the object EnableOperatingMode is only relevant in the case of the QAX33.1 or QAX34.1, as only these two room units have the option of selecting Fan1, Fan2 or Fan3.

Where QAX units without an address switch are still in use, only one room unit per automation station can be integrated. The room unit number in such cases is then "1".

LonWorks addressing

There are two ways to integrate data points from LonWorks devices:

- via Discipline I/O
- via standard inputs/outputs (the latter approach is only sensible with a small number of data points to be integrated, e.g., from third-party devices)

Address syntax

The block registers the control variables and output variables of the RX devices (outside the CFC chart) in accordance with the information in the [IOAddr] property (Input/output address).

Addressing starts with the prefix "L=".

Addressing via discipline I/O

L=DeviceType DeviceNo. GroupIndex(MappingTableNo.)

- DeviceType: M (Master), S (Slave)
- DeviceNo: Field device identification number
- GroupIndex: Group identification: Up to 4 similar groups of an application unit may exist in the field device (e.g., lighting or window-blind groups). The group index number is optional.
- MappingTableNo: Number of the mapping table which is valid for that Discipline I/O.

More than one device can be specified for each [IOAddr] string. The devices are separated with a semicolon. However, the maximum [IOAddr] string length of 60 characters must not be exceeded.

Addressing via standard I/O

L= DeviceType DeviceNo.GroupIndex(3RD[NVIndex.FieldIndex])

- DeviceType: M (Master). There are no slaves (S) with third-party devices There is only ever one device.
- DeviceNo: Field device identification number
- GroupIndex: Group identification: Up to 4 similar groups of an application unit may exist in the field device (e.g., lighting or window-blind groups). The group index number is optional.
- ObjectType: Constant for third-party devices: 3RD.
- NVIndex: Network variable referenced in the third-party device.
- FieldIndex: Element number, if the network variable is structured

KNX addressing

You can integrate data points from KNX devices as follows:

- See *PX KNX, RXB integration - S-Mode* (CM1Y9775)
- See *PX KNX, RXB/RXL integration - Individual addressing* (CM1Y9776)
- Address Info LED for PX KNX: D=1001

19.7 Discipline I/Os

Discipline I/Os are standardized combinations of inputs and outputs related to a specific application. They have a predefined number of parameters.

Three different input variable types can be interconnected to Discipline I/Os:

- Simple value
- Trigger value
- Commandable value

Simple value

The input value can be connected via the data flow. In the engineering tool, this is preceded by a function block or compound, e.g., a Scheduler. However, if the input value is not connected, it can also be modified via BACnet client. The subsystem registers a change in the input value by comparing the value with the process image and transferring it to the field devices.

Trigger value

This input value is the logical image, or memory map, of an analog positioning command and describes its properties. Within the program, the Present Value is made available to the block as a program value. The block transfers the program value to the subsystem, from where it is transmitted to the field device.

Writing to this value acts as a trigger. This makes it possible, e.g., to generate the output of the same value (e.g., Lighting 100%, followed later by 100% again). In this case the subsystem registers the trigger value and transmits the value to the devices. This capability is required when the same variable can be modified from several sources (e.g., when Desigo CC writes 100.0%, the local operator unit writes 0.0% and the Desigo CC user wants to rewrite the value of 100.0%). The sources can be BACnet clients or system function blocks.

Only analog trigger values may be used.

Commandable value

The input value is the logical image, or memory map, of an analog positioning command and describes its properties. Within the program, the Present Value is made available to the block as a program value. The block transfers the program value to the subsystem, from where it is transmitted to the field device.

The commandable value is based on the BACnet priority-mechanism (which is the same as for the output blocks – refer to Section 0). A commandable value can be operated from various sources. Each source has its own priority. The sources are mutually exclusive (interlock). The source with the highest priority prevails, e.g., Emergency = Priority 1, Façade control = Priority 6, Operator = Priority = 8). The sources can be BACnet operator units or system function blocks (grouping function).

Only analog commandable values can be used.

19.8 Reliability table

| Value (decimal) | Text |
|-----------------|--|
| 0 | No error recognized. |
| 1 | No sensor. |
| 2 | Above the range. |
| 3 | Below the range. |
| 4 | Continuous loop. |
| 5 | Short circuit. |
| 6 | No output. |
| 7 | Unreliable other. |
| 8 | Process error |
| 9 | Multistate fault. |
| 64 | Subsystem not supported. |
| 65 | Subsystem feedback not supported. |
| 66 | Invalid address (syntax error). |
| 67 | Invalid feedback address (syntax error). |
| 68 | Invalid address value. |
| 69 | Invalid feedback address value. |
| 70 | Invalid address parameter (syntax error). |
| 71 | Invalid feedback address parameter (syntax error). |
| 72 | Invalid address parameter value. Unsupported signal types in the automation station also generate reliability error message 72. |
| 73 | Invalid parameter value for feedback address. |
| 74 | Destination device unknown. |

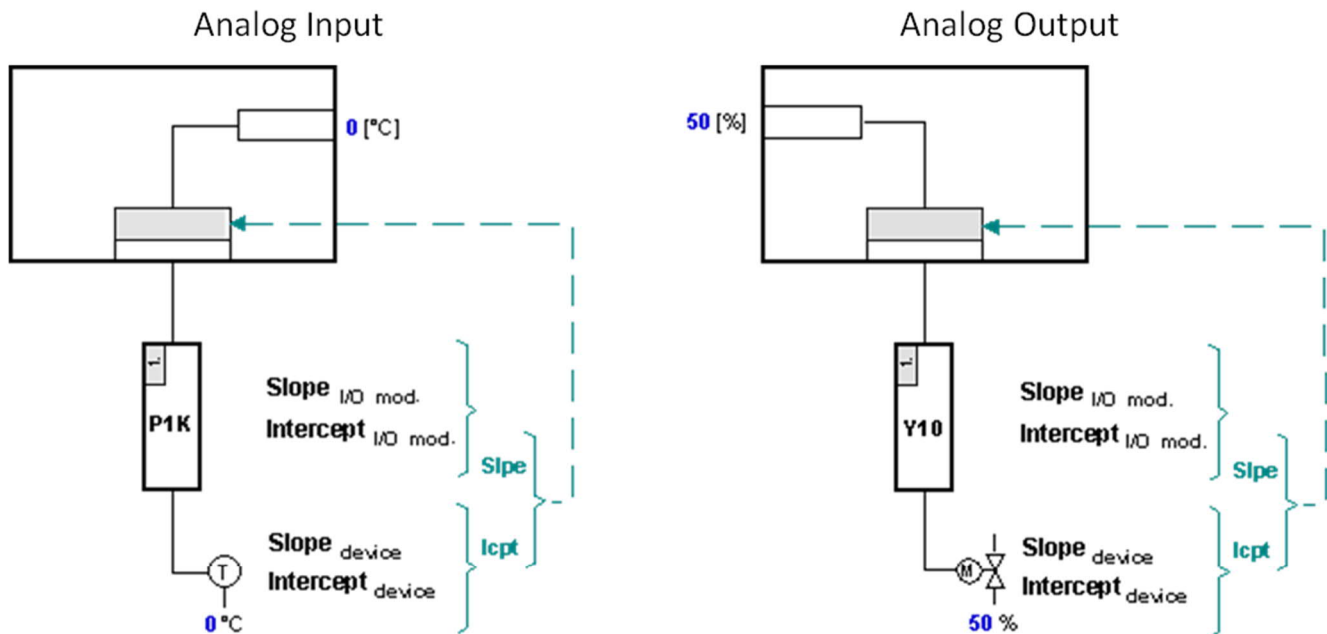
| Value (decimal) | Text |
|-----------------|---|
| 75 | Feedback device unknown. |
| 76 | Destination object unknown. |
| 77 | Feedback object unknown. |
| 78 | Unsuitable destination type. |
| 79 | Unsuitable feedback type. |
| 80 | Unreliable destination object. |
| 81 | Unreliable feedback object. |
| 82 | Invalid subsystem (syntax error). |
| 83 | Invalid feedback subsystem (syntax error). |
| 84 | Memory full. |
| 85 | Unreliable target device. |
| 86 | Communication failure reported in subsystem. |
| 87 | Alarm in subsystem application. |
| 88 | Maximum BACnet references reached for device. |
| 89 | Reliable participant. |
| 90 | Feedback error reported in binary output. |
| 91 | Invalid reference: Address not valid. |
| 92 | Reference object cannot be commanded. |
| 93 | Actual operating mode not found in command list. |
| 94 | Invalid priority set for command (valid : 2,4,14,16). |
| 95 | Invalid object number configured in sequence table. |
| 96 | Invalid object type configured in sequence table. |
| 97 | Invalid step control configured in sequence table. |
| 98 | Neighboring object not reachable. |
| 99 | Command lists indicate different variables. |
| 100 | Invalid calendar reference. |
| 101 | Configured switch kind not supported by destination controller. |
| 102 | Multistate mapping error. |

19.9 Slope [Slpe] and Intercept [Icpt]

[Slpe] and [Icpt] value exist for:

- I/O modules (PX Modular and PX Compact)
These values impact signal type (the I/O module).
- Siemens field devices
These values affect the combination of Slpe and Icpt values for the signal type, the field device and its measurement and positioning range. XWP automatically enters these values, and they can be changed there, e.g., to consider the line resistance of a sensor or to describe a third-party sensor.
- BACnet referencing
- PPS2 interface

The combined values [Slpe] and [Icpt] can be calculated as follows from individual values for signal type (I/O module) and characteristic curve (field device):



$$Slpe = \frac{Slope_{I/O-Mod.}}{Slope_{Device}}$$

$$Icpt = \frac{Intercept_{Device}}{Slope_{Device}} + Intercept_{I/O-Mod.}$$

$$Slpe = Slope_{I/O-Mod.} \times Slope_{Device}$$

$$Icpt = Slope_{I/O-Mod.} \times Slope_{Device} \times Intercept_{Device}$$

$$+ Slope_{I/O-Mod.} \times Intercept_{I/O-Mod.}$$

Siemens Building Technologies field devices: XWP automatically enters the combined values [Slpe] and [Icpt] (for the signal type, the field device and its measurement or positioning range) on the I/O block.

Third-party field devices: You can calculate the value [Slpe] and [Icpt] using the Intercept Calculator.

[Slpe] and [Icpt] Analog Input

TX- and PT-I/O modules PX modular

In the Desigo PX modular automation stations, the analog input block is used with the following TX-I/O and PT-I/O modules:

| Signal type measurement | Description | Standard measuring range | Resolution on the bus | Value range on the bus | [Slpe] | [Icpt] |
|-------------------------|---------------|--------------------------|-----------------------|------------------------|--------|--------|
| R1K | LG-Ni 1000 | -50 ... 150 °C | 1/100 °C | -5000 ... 15000 | 0.01 | 0 |
| P100 | Pt100 | 0 ... 250 Ohm | 1/100 Ohm | 0 ... 25000 | 0.01 | 0 |
| R250 | Resistance | 0 ... 250 Ohm | 1/100 Ohm | 0 ... 25000 | 0.01 | 0 |
| Pt100_4 | Pt100 | -50 ... 600 °C | 1/100 °C | -5000 ... 40000 | 0.01 | 0 |
| P1K | Pt1000 | 0 ... 2 500 Ohm | 1/10 Ohm | 0 ... 25000 | 0.1 | 0 |
| R2K5 | Resistance | 0 ... 2 500 Ohm | 1/10 Ohm | 0 ... 25000 | 0.1 | 0 |
| T1 | PTC sensor | -50 ... 150 °C | 1/100 °C | -5000 ... 15000 | 0.01 | 0 |
| Ni1K | LG-Ni 1000 | -50 ... 180 °C | 1/100 °C | -5000 ... 18000 | 0.01 | 0 |
| Pt1K375 | Pt1000 (NA) | -50 ... 180 °C | 1/100 °C | -5000 ... 18000 | 0.01 | 0 |
| Pt1K385 | Pt1000 (EU) | -50 ... 600°C | 1/100 °C | -5000 ... 60000 | 0.01 | 0 |
| NTC10K | NTC sensor | -40 ... 115 °C | 1/100 °C | -5000 ... 11500 | 0.01 | 0 |
| NTC100K | NTC sensor | -40 ... 125 °C | 1/100 °C | -5000 ... 12500 | 0.01 | 0 |
| U10 | DC 0 ... 10V | 0 ... 10 Volt | 1/1000 V | 0 ... 10000 | 0.001 | 0 |
| I420 | DC 4 ... 20mA | 4 ... 20 mA | 1/1000 mA | 4000 ... 20000 | 0.001 | 0 |
| I25/020 (Shunt 200 Ohm) | DC 0 ... 25mA | 1 ... 5 mA | 1/1000 V | 0 ... 5000 | 0.001 | 0 |

| Signal type measurement | Description | Standard measuring range | Resolution on the bus | Value range on the bus | [Slpe] | [Icpt] |
|-----------------------------|-----------------|--------------------------|-----------------------|------------------------|---------|--------|
| I25/020 (Shunt 100 Ohm) | DC 0 ... 25mA | 0 ... 10 mA | 1/500 V | 0 ... 5000 | 0.002 | 0 |
| I25/020 (Shunt 50 Ohm) | DC 0 ... 25mA | 0 ... 20 mA | 1/250 V | 0 ... 5000 | 0.004 | 0 |
| I25/020 (Shunt 40 Ohm) | DC 0 ... 25mA | 0 ... 25 mA | 1/200 V | 0 ... 5000 | 0.005 | 0 |
| I25/020 TX-I/O* | DC 0 ... 20mA*) | 0 ... 20 mA* | 1/1000 mA | 0 ... 20000 | 0.001* | 0 |
| U10 (Shunt 400 Ohm) TX-I/O* | DC 0 ... 25mA*) | 0 ... 25 mA* | 1/1000 V | 0 ... 10000 | 0.0025* | 0 |

Key

* TX-I/O modules support only 0 ... 20 mA. For a range of 0 ... 25 mA, use the shunt for 400 Ohm (0.1%, 1 W) and measure the voltage with U10.

I/O configuration PX Compact

The analog input block is used in the Desigo PX Compact PXC10 TL to PXC52 automation station in cases where an LG Ni1000 sensor (signal type R1K) or DC 0...10 V (U10) is connected to device terminals X1...X16 of Module 001.

The following information results:

| Signal type measurement | Description | Standard measuring range | [Slpe] | [Icpt] |
|-------------------------|-------------|--------------------------|--------|--------|
| R1K | LG-Ni 1000 | -50...150 °C | 0.01 | 0 |
| U10 | DC 0...10V | 0...10 Volt | 0.001 | 0 |

BACnet referencing

Reference to a value in another BACnet object. As the referenced value is already available as a converted or resulting value, no conversion is required, that is, [Slpe] must be defined as 1 and [Icpt] as 0.

PPS2 interface

The measured value from a room unit connected via the PPS2 interface. In the analog input block, only Objects 24 (setpoint correction) and 40 (room temperature) may be used. As the value is already available as a converted or referenced value, no conversion is required, that is, [Slpe] must be defined as 1 and [Icpt] as 0.

[Slpe] and [Icpt] Analog Output

I/O modules PX modular

In the PX modular automation stations, the analog output block is used with the following signal types:

| Signal type positioning | Description | Standard measuring range | [Slpe] | [Icpt] |
|-------------------------|--------------|--------------------------|--------|--------|
| Y10S | DC 0...10 V | 0 ... 10 V | 100 | 0 |
| Y420 | DC 4...20 mA | 4 ... 20 mA | 160 | 4000 |
| Y250T (P-bus)* | Three-point | AC 24...250 Volt | 2.55* | 0 |
| Y250T (Island bus)* | Three-point | AC 24...250 Volt | 100* | 0 |

Key

* Value [Slpe] for Y250T is not a physical value, but rather a special code controlling output of the AO to two relay outputs. This code differs between P-bus and island bus.

I/O configuration PX Compact

The analog output block is used in the PX compact automation stations, when valves or actuators with DC 0...10 V control signals, signal type Y10S, are connected to device terminals Y1...Y8 of Module 004.

| Signal type positioning | Description | Standard measuring range | [Slpe] | [Icpt] |
|-------------------------|-------------|--------------------------|--------|--------|
| Y10S | DC 0...10 V | 0 ... 10 V | 1000 | 0 |

PPS2 interface

Transfer of an analog control command to a room unit connected via the PPS2 interface. Only Object 195 (= Room temperature display) can be used in the analog output block. As the value is already available as a converted or referenced value, no conversion is required, that is, [Slpe] must be defined as 1 and [Icpt] as 0.

Line resistance with [Icpt]

For analog inputs (measurement of temperatures or resistances), most signal types are calibrated at a line resistance of 1 Ohm. The [Icpt] can be changed at the AI block if the line resistance deviates strongly from 1 Ohm.

Measuring resistances (internal resolution = 1/10 Ohm):

| Line resistance | | [Slpe] | | [Icpt] |
|----------------------------|-----------|-------------|---|-----------|
| P1K (Pt1000) | | | | |
| 0 Ohm | 0.0259740 | -259.480519 | 0 | 0.259740 |
| Default = 1 Ohm | 0.0259740 | -259.740260 | 0 | 0 |
| 2 Ohm | 0.0259740 | -260.000000 | 0 | -0.259740 |
| 3 Ohm | 0.0259740 | -260.259740 | 0 | -0.519481 |
| R2K5 | | | | |
| P1K (0...2500 Ohm) | | | | |
| 0 Ohm | 0.1 | 1 | 0 | 1 |
| Default = 1 Ohm | 0.1 | 0 | 0 | 0 |
| 2 Ohm | 0.1 | -1 | 0 | -1 |
| 3 Ohm | 0.1 | -2 | 0 | -2 |
| R250 | | | | |
| 0 Ohm | 0.01 | 1 | 0 | 1 |
| Default = 1 Ohm | 0.01 | 0 | 0 | 0 |
| 2 Ohm | 0.01 | -1 | 0 | -1 |
| 3 Ohm | 0.01 | -2 | 0 | -2 |
| R250 | | | | |
| P100 (0...250 Ohm)* | | | | |
| 0 Ohm | 0.01 | 0 | 0 | 0 |
| Default = 1 Ohm | 0.01 | -1 | 0 | -1 |
| 2 Ohm | 0.01 | -2 | 0 | -2 |
| 3 Ohm | 0.01 | -3 | 0 | -3 |

Key

| | | |
|--|---|---|
| PT-I/O modules | P100 is a four-wire type | Default line resistance = 0 Ohm Line resistance not compensated |
| TX-I/O modules with island bus integration | Pt100_4 is a four-wire type R250 is a two-wire type | Default line resistance = 0 Ohm Line resistance not compensated Default line resistance = 1 Ohm |
| TX-I/O modules with BIM integration | Pt100_4 is a four-wire type R250 is a two-wire type, but must be connected to four terminals using bridges | Default line resistance = 0 Ohm Line resistance not compensated Default line resistance = 0 Ohm |

Measuring temperature (internal resolution = 1/100 °C):

| Line resistance | | [Slpe] | [lcpt] |
|------------------|------|--------------|---------------------------|
| Pt 1K 385 | | 3.85 | 0.259740 |
| 0 Ohm | 0.01 | 0.259740 | |
| Default = 1 Ohm | 0.01 | 0 | |
| 2 Ohm | 0.01 | -0.259740 | |
| 3 Ohm | 0.01 | -0.519481 | |
| Pt 1K 375 | | 3.75 | 0.266667 |
| 0 Ohm | 0.01 | 0.266667 | |
| Default = 1 Ohm | 0.01 | 0 | |
| 2 Ohm | 0.01 | -0.266667 | |
| 3 Ohm | 0.01 | -0.533333 | |
| Ni1K | | 5 | 0.2 |
| 0 Ohm | 0.01 | 0.2 | |
| Default = 1 Ohm | 0.01 | 0 | |
| 2 Ohm | 0.01 | -0.2 | |
| 3 Ohm | 0.01 | -0.4 | |
| T1 | | 9.57 | 0.104450 -50...0 °C |
| | | 10.39 | 0.096246 0...50 °C |
| | | 11.31 | 0.088417 50...100 °C |
| | | 12.36 | 0.080893 100...150 °C |
| 0 Ohm | 0.01 | 0.096246 | |
| Default = 1 Ohm | 0.01 | 0 | |
| 2 Ohm | 0.01 | -0.096246 | |
| 3 Ohm | 0.01 | -0.192493 | |

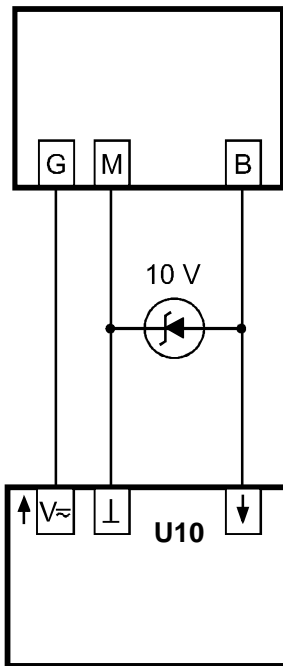
Pt100_4

Pt100 is a four-wire type, default line resistance = 0 Ohm
-> Line resistance is not compensated

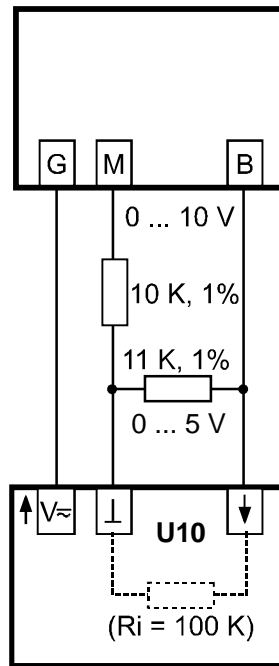
Power surges on U10 inputs

The U10 inputs are designed for DC 0 ... 10 V with a narrower high / low tolerance range. The input reports an error when a value is stored that outside this range. A transient voltage suppressor can prevent an error message. A faulty response from the analog signal supplied by the automation station can no longer be detected.

Solution examples:

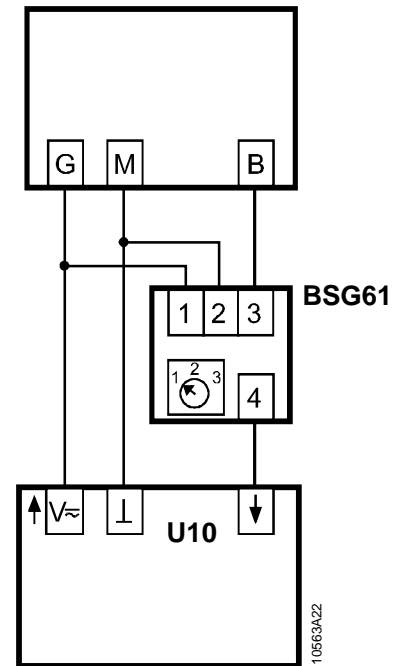


Zener diode



Voltage divider

Slope must be adapted to 0...5 V
(0.01 -> 0.005)
Precision resistance, e.g., VISHAI
MBB/SMA 0207



Active setpoint adjuster BSG61
(Datasheet CE1N1992)

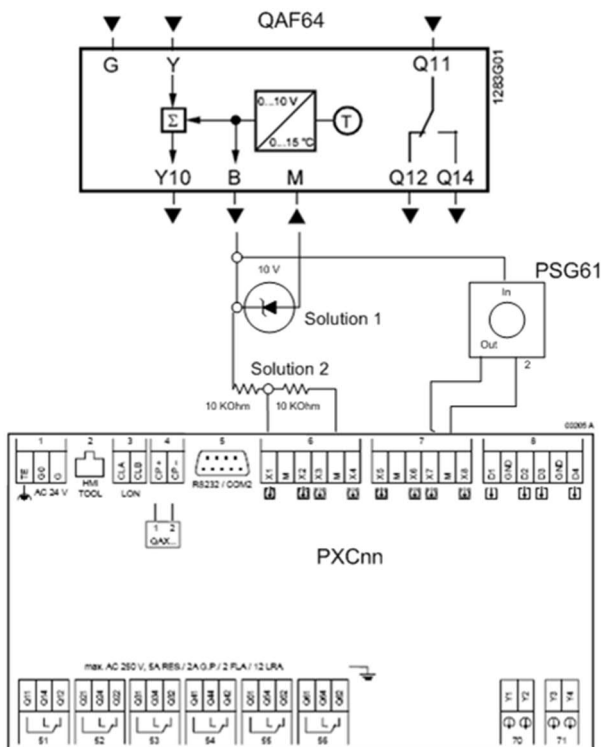
Switch position 1 (Setpoint limit
control) Setpoint 100%

[Icpt] and [Slpe] for BT devices

Note for all U10 inputs

The physical inputs are designed for 0 -10V with narrow high and low tolerance limits. If a value falls outside this range, the input transmits an error signal. However, provided it is established that the peripheral devices are in order, an error signal can be prevented by using a transient voltage suppressor (10 V Zener diode and two resistors). A faulty response from the analog input signal supplied cannot subsequently be detected in the automation station.

Example of a circuit including the QAF64 which transmits more than 10 volts



19.10 Addressing entries for PXC...-U, PTM and P-Bus

Addressing entries PX modular (PXC...-U)

For the PX modular series, the P bus I/O modules at the Input-Output address pin [IOAddr] start with the prefix: "P=".

Address syntax: P= Module.Channel (Signal type, parameter)

Example: P=2.1 (Y10S,15)

The exception is the Info LED which must have the prefix "C=" because the fixed address 8.1, which is used for the Info LED may also be used by an I/O module.

Info-LED for PX KNX: D=1001.

Address entries required when using the modular series automation station in conjunction with TX-I/O modules

| Type | Module addressing | I/O point or channels |
|---------------|-------------------|-----------------------|
| Desigo TX-I/O | 1...120 | 1...16 |
| Desigo PT-I/O | 1...255 | 1...8 |
| PX Info LED | 8 | 1 |

| Module type | Signal type | Parameters | Example |
|---------------|--|--------------------|---------------------------------------|
| Analog Input | R1K, P1K, U10, I25, I420 P100 T1 (only TX-I/O) | - | P=1.1 (R1K) |
| | AI, AIS, AIL, AISL | - | P=2.1 (AIS) |
| Analog Output | Y10S | NO, KEEP 0...30 | P=2.1 (Y10S, KEEP) P=2.1 (Y10S,15) |
| | Y250T | 1...13, 1...13 | P=3.1 (Y250T,8) P=3.1 (Y250T,8,10) |
| | Y420 | - | P=34.1 (Y420) |

| Module type | Signal type | Parameters | Example |
|-----------------------------|--|-------------------------------|-------------------------------------|
| | <i>AO, AOS, AOSL, AOL</i> | | P=36.1 (AOS) |
| Binary Input | D20, D20S D42, D250 (only PT-I/O) | - | P=25.2 (D20) |
| | <i>DI, DIS, DIL, DISL</i> | - | P=26.3 (DIS) |
| Counter Input | C | - | P=38.1 (C) |
| Info LED | Q_LED PX KNX: D=1001 | - | C=8.1(Q_LED) |
| Binary Output | Q250_P, Q250A_P | 0, 1...600 | P=12.1 (Q250_P) |
| | Q250 QD, Q250B, (only PT-I/O) | - | P=1.1 (QD) P=14.1 (Q250) |
| | <i>DO, DOS, DOL, DOSL</i> | - | P=15.2 (DOS) |
| Multistate Input | D20 D42, D250 (only PT-I/O) | Binary - Mapping | P=1.1;1.2 (D20) |
| | | Updown - Mapping | |
| | | 1:n - Mapping | |
| Multistate Output | <i>DI, DIS, DIL, DISL</i> Q250 Q250B, QD (only PT-I/O) | Binary - Mapping | P=7.1 (DIS) P=1.1;1.2;1.3 (Q250) |
| | | Updown - Mapping | |
| | | 1:n - Mapping | |
| | Q250_P3 | 0, 1...600 | P=1.1 (Q250_P3,120) |
| Q-M3 QD-M2 (only PT-I/O) | - | P=1.1 (Q-M2) P=1.1 (QD-M2) | |
| | <i>DO, DOS, DOL, DOSL</i> | - | P=26.3 (DIS) |

Signal types shown in *italics* are used to map virtual modules for use with I/O OPEN at module level. Signal types AIS, AOS, DIS and DOS deliver a 16 bit value with status information, while signal types AISL, AOSL, DISL and DOSL deliver a 32 bit value with status information. All other signal types deliver a 16/32 bit value without status information.

While all the module types listed may be connected to any P-bus addresses, not all module types have 16 channels.

Parameter values

Parameter values for the analog output, binary output and multistate output blocks:

Y10S

Failsafe function (emergency control function) if the transfer of data over the P-bus fails (for longer than 4 seconds) or in the event of a power failure. (an operating voltage of AC 24 V must be available).

NO -> Module output signal goes to 0 V.

KEEP -> Module output signal remains at previous value.

0...30 -> Module output signal 0 = 0 V, 1 = 0.33 V, etc. , ... 30 = 10 V.

Y250T

1...13, 1...13 Runtime ranges for On/Off signals (the ranges do not need to be the same for On/Off). Values 1...13 correspond to the following runtimes:

1 = 8.5 ...13 seconds

2 = 13 ... 18 seconds

3 = 18 ...25 seconds

4 = 25 ...35 seconds

5 = 35 ... 48 seconds

6 = 48 ... 66 seconds

7 = 1.1 ... 1.6 minutes

8 = 1.6 ... 2.3 minutes

- 9 = 2.3 ... 3.2 minutes
- 10 = 3.2 ... 4.5 minutes
- 11 = 4.5 ... 6.3 minutes
- 12 = 6.3 ... 9.0 minutes
- 13 = 9.0 ... 11 minutes

The PTM1.2Y250T(-M) module can only implement one runtime. It therefore uses the opening-command runtime for closing commands.

Q250_P, Q250A_P, Q250_P3

0, 1...600 -> Pulse times, where 0 = 0.5 seconds and then 1 = 1 second, 2 = 2 seconds etc. up to 600 (=600 seconds).

Pulse times for island bus applications:

Values in the I/O address editor: 0...255 (corresponds to 0...25.5 seconds)

Default = 5 (corresponds to 0.5 seconds).

Addressing entries PX Compact (PXC...)

The addressing procedure is almost identical for Desigo PX compact and for Desigo PX modular. However, the valid address ranges and signal types are not the same as those used for the addressing of individual P-bus I/O modules.

For PX compact, the on-board I/O modules at the [IOAddr] pin start with a "C" (prefix: "C=").

Address syntax: C=Module.Channel (Signal type, parameter)

Example:C=2.1 (Y10S, NO)

The table below shows the available address ranges and signal types, which vary according to the Desigo PX compact automation station (each with its own integrated, fixed configuration of I/Os) type.

| Desigo PX compact | PXC10-TL ¹ | | PXC12 PXC12-T | | PXC22 PXC22-T | | PXC36 PXC36-T | | PXC52 | | Signal type |
|---|-----------------------|---------|---------------|------------------|---------------|------------------|---------------|--------------------|--------|--------------------|-------------------------------|
| | Module | Channel | Module | Channel | Module | Channel | Module | Channel | Module | Channel | |
| Uni-versal Inputs (UI: for AI, DI) | 1 | 1...4 | 1 | 1...6 X1...X6 | 1 | 1...8 X1...X8 | 1 | 1...12 X1...X12 | 1 | 1...16 X1...X16 | R1K, U10, D20 T1, P1K, N1K |
| Digital Inputs (DI) (Counter Input) | 2 | 1...4 | - | - | 2 | 1...4 D1...D4 | 2 | 1...4 D1...D4 | 2 | 1...4 D1...D4 | D20, C |
| Digital Inputs (DI) | 3 | 1...4 | - | - | - | - | 3 | 1...8 D5...D12 | 3 | 1...12 D5...D16 | D20 |
| Analog Outputs (AO) | - | - | 4 | 1...4 Y1...Y2 | 4 | 1...4 Y1...Y4 | 4 | 1...6 Y1...Y6 | 4 | 1...8 Y1...Y8 | Y10S |
| Digital Outputs (DO) | 5 | 1...2 | 5 | 1...2 51...54 | 5 | 1...6 51...56 | 5 | 1...8 51...58 | 5 | 1...12 51...62 | Q250 |
| Manual switch ² (only PXC36-S) | - | - | - | - | - | - | 7 | 1...4 | - | - | D_M3 |
| LEDs | 8 | 2...5 | - | - | - | - | 8 | 2...7 | - | - | Q_LED |

| Desigo PX compact | PXC10-TL ¹ | | PXC12 PXC12-T | | PXC22 PXC22-T | | PXC36 PXC36-T | | PXC52 | | Signal type |
|--|-----------------------|---------|---------------|---------|---------------|---------|---------------|---------|--------|--|--|
| | Module | Channel | Module | Channel | Module | Channel | Module | Channel | Module | Channel | |
| Info LED | 8 | 1 | 8 | 1 | 8 | 1 | 8 | 1 | 8 | 1 | Q_LED |
| PPS-2 signal ³ | 3 | 1..5 | 3 | 1..5 | 3 | 1..5 | 3 | 1..5 | 3 | 1..5 | R1K, U10, D20 |
| PXC52 from Desigo V5 ⁴ : Universal Inputs / Outputs | | | | | | | | | 1 4 | 1...16 X1...X16 1...8 Y1...Y8 | D20, C R1K, U10, D20 T1, P1K, N1K, Y10S |

Key

- 1 For PXC10-TL the two Alarm1/2 buttons and the DIL switches1/2 are mapped to the modules with the Address 3.
- 2 The manual switch can only be loaded into the application if the DIL switches (in the cover of the PXC36-S) are set correctly.
- 3 Syntax for PPS-2 signal: Q=Romm device number.Object number (profile number). Up to five devices can be connected.
- 4 The current UI and AO can all be configured as AI, DI, CI, or AO.

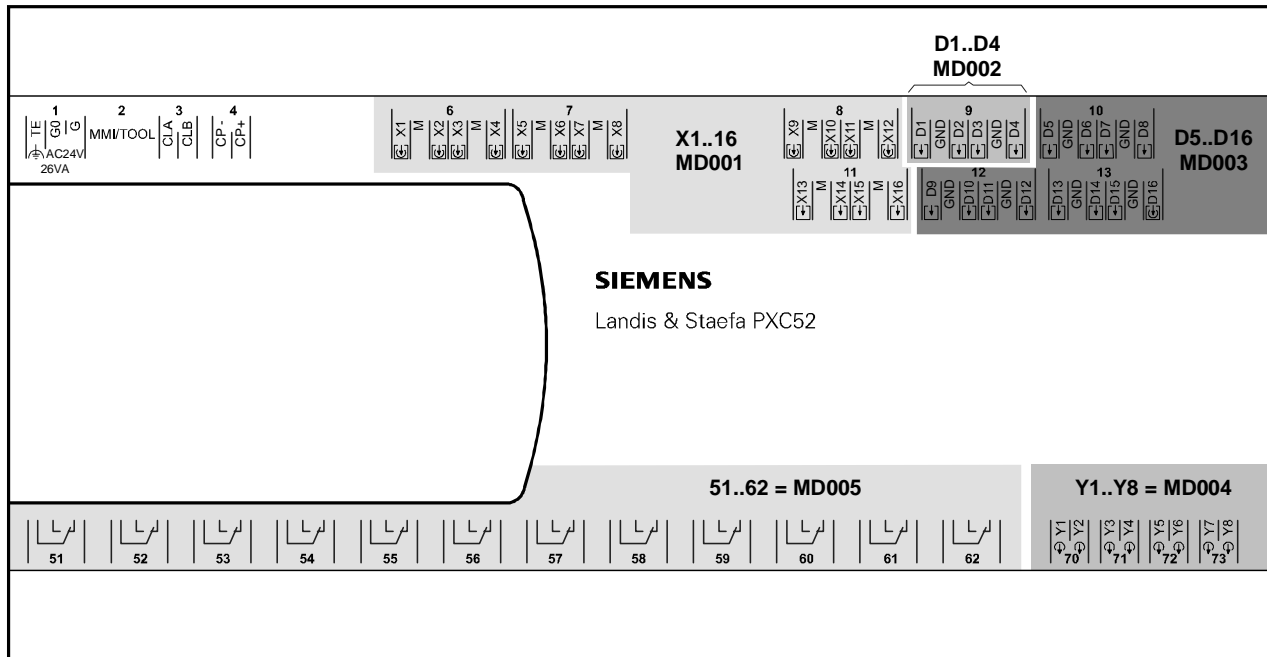
Signal type if no application is loaded (wiring test): X1...X16 = Y10S, Y1...Y8 = R1K.

Module 4

For Module 4, the universal outputs (UO for AO and DO) not only control proportional actuators (AO), but can also be used as binary switch commands (DO).

- Analog Output = 0...10 V
- Binary Output = DC 0 or 24 V, max. 22 mA with the use of an additional external relay.

Layout of PXC52 housing with address ranges



Addressing multistate I/Os with PTM

Multistate input

The multistate value is made up of several separate binary measured values.

Addressing is via the input/output address [IOAddr]. In both the modular and the compact series, the logical and physical I/O must be located in the same automation station, but they do not need to be contiguous. The addressing cannot extend across automation stations. The addresses must be on the same module for TX-I/O.

Simple mapping

Syntax: P=Module.Channel;Module.Channel;Module.Channel;Module.Channel (Signal type)

Examples:

- P=1.1 (D20)
- P=1.1;1.2 (D20)
- P=1.1;1.2;1.3 (D20)
- P=1.1;1.2;1.3;1.4 (D20)
- P=10.3 (DIS)

Up to four binary status values (e.g., Off/St1/St2/St3/St4) can be registered. The signals to be registered, which are addressed via Module.Channel, must always be of the same hardware signal type. With the simple mapping procedure, to enable the multistate input to interpret the current binary signals correctly, only one binary signal may be present at any one time. If several binary signals are present at once, this is displayed as an error at the [Rlb] pin.

The examples below show a possible application for multistate input blocks in conjunction with the physical I/O modules. The example on the left of the diagram is a multiple I/O module, while the one on the right shows the mapping of several individual I/O modules in one multistate input block.

Multistate output

The multistate value from the program is converted in the Multistate Output block into a switching command. Addressing is via [IOAddr]. For PX modular, the syntax is as follows:

Syntax: P=Module.Channel;Module.Channel;Module.Channel;Module.Channel (signal type, parameter)

Examples:

- P=1.1 (Q250)
- P=1.1;1.2 (Q250)
- P=1.1;1.2;1.3 (Q250)
- P=1.1;1.2;1.3;1.4 (Q250)
- P=10.1 (Q250-P3,120)
- P=24.7 (DOS)

Values with up to four stages can be processed. The signals to be registered, which are addressed via Module.Channel, must always be of the same hardware signal type. In the case of a multistate output on the hardware side, there is one address only (this is only possible with PXC modular automation stations).

Error handling

If an automation station does not support a given address (e.g., incorrect syntax) or a given I/O system, this will lead to a reliability error, which will be displayed at the [Rlb] pin.

Advanced mapping (Multistate Input)

The manual switch can be encoded on the PX compact in various ways, e.g.:

- (Auto/Off/On) or (Off/Auto/On)
- (Auto/Off/S1/S2) or (Off/Auto/S1/S2)

To avoid having to keep adapting the data types and text groups in the system, the manual switch must always be represented in the same way within the system:

- (Auto/Off/On)
- (Auto/Off/S1/S2)

A prerequisite for this approach is that it must be possible in the multistate input block to configure the hardware coding and mapping to the standardized manual switch. This is made possible with parameters in the address.

1_n-Mapping (Multistate Input and Output)

Syntax: P=Module.channel;Module.channel;Module.channel;Module.channel (signal type, a,b,c,d,e)

a represents [PrVal] for HW-I/O (0,0,0,0)

b represents [PrVal] for HW-I/O (1,0,0,0)

c represents [PrVal] for HW-I/O (0,1,0,0)

d represents [PrVal] for HW-I/O (0,0,1,0)

e represents [PrVal] for HW-I/O (0,0,0,1)

Example: P=1.1;1.2;1.3;1.4 (D20, **1, 3, 2, 4, 5**)

| [PrVal] | Addr1 | Addr2 | Addr3 | Addr4 | Comment / Text group |
|---------|-------|-------|-------|-------|----------------------|
| 1 | 0 | 0 | 0 | 0 | Auto |
| 3 | 1 | 0 | 0 | 0 | Stage 1 |
| 2 | 0 | 1 | 0 | 0 | Off |
| 4 | 0 | 0 | 1 | 0 | Stage 2 |
| 5 | 0 | 0 | 0 | 1 | Stage 3 |

UpDown mapping (Multistate Input and Output)

Application: Connecting/disconnecting further stages.

Example: Electric heating registers, multi-stage burners.

Syntax: P=Module.channel;Module.channel;Module.channel;module.channel (signal type, UPDOWN)

With "Up/Down" mapping, more than one hardware input or output may be active.

Binary mapping (Multistate Input and Output)

Application: Output of an integer in binary form.

Example: Binary electric heating coil.

Syntax: P=Module.channel;Module.channel;Module.channel;Module.channel (signal type, BINARY)

With binary mapping, more than one hardware input or output may be active.

20 Room automation

Desigo room automation

Desigo room automation offers solutions with greater functionality and flexibility allowing for energy-optimized plant operation without loss of comfort (efficiency class A).

The DXR1 and DXR2 room automation stations are perfectly suited to exclusively automate heating, ventilation, and air conditioning in a room. In addition, the DXR2 can be extended with lighting and shading functions by adding devices with KNX PL- Link.

The PXC3 modular room automation stations are used in buildings with multiple disciplines for room automation (HVAC, lighting, blinds) all combined in one system.

Desigo RX

Desigo RX is a proven room automation product range featuring comprehensive communications and application functions for individual rooms. The product range consists of a series of communicating room controllers RXB with operator units and predefined applications for HVAC, lighting, and blinds. Desigo RX room automation is capable of autonomous operation. Integration of LonWorks or KNX network via the system controllers provides for additional functionality.

20.1 Desigo room automation

New guidelines to save energy and lower operating costs and greater requirements for comfort and design require a more sophisticated interaction between a building's various technical installations. The modular and compact room automation stations combine lighting, shading, and HVAC in one total solution and provide a direct connection to the automation stations via BACnet.

Overview of room automation stations

| Product range | Configurable | | Programmable | |
|--|--|--|---------------------------------------|--------------|
| Applications and tool | Configurable with ABT Site | | Programmable with library in ABT Pro | |
| Communication (Backbone) | BACnet/IP | BACnet MS/TP | BACnet/IP | BACnet MS/TP |
| Communication with sensors and actuators in room (integration) | KNX PL-Link ¹ | KNX PL-Link ² | KNX PL-Link KNX (with ETS) DALI | KNX PL-Link |
| System integration/functions | PXG3.L | PXG3.M | PXG3.L | PXG3.M |
| Modular controller I/Os | | | PXC3.E.. TXM | |
| Compact controller | DXR2.E.. DXR1.E.. | DXR2.M.. DXR1.M.. | DXR2.E.. | DXR2.M.. |
| DALI extension | | | PXC3.E16A PXC3.E..A | |
| Communication with room units | KNX PL-Link | KNX PL-Link | KNX PL-Link | KNX PL-Link |
| Room units | QMX3.. ³ QMX1.. ⁴ | QMX3.. ³ QMX1.. ⁴ | QMX3.. | QMX3.. |
| Touch screen | | | QMX7.. | |

Key

- ¹ DXR1.E..: Applicable for some variants only.
- ² Not applicable for DXR1.M...
- ³ Only for DXR2...
- ⁴ Only for DXR1...

KNX PL-Link

KNX PL-Link (Peripheral Link) connects communicating room and field devices (room devices, sensors, actors) with the room automation station.

DALI

DALI (Digital Addressable Lighting Interface) helps control lighting.

20.1.1 Configurable

DXR2.. can automate up to two rooms for heating, ventilation, air conditioning, shading, and lighting, whereas DXR1 can automate one room and heating, ventilation, and air conditioning only.

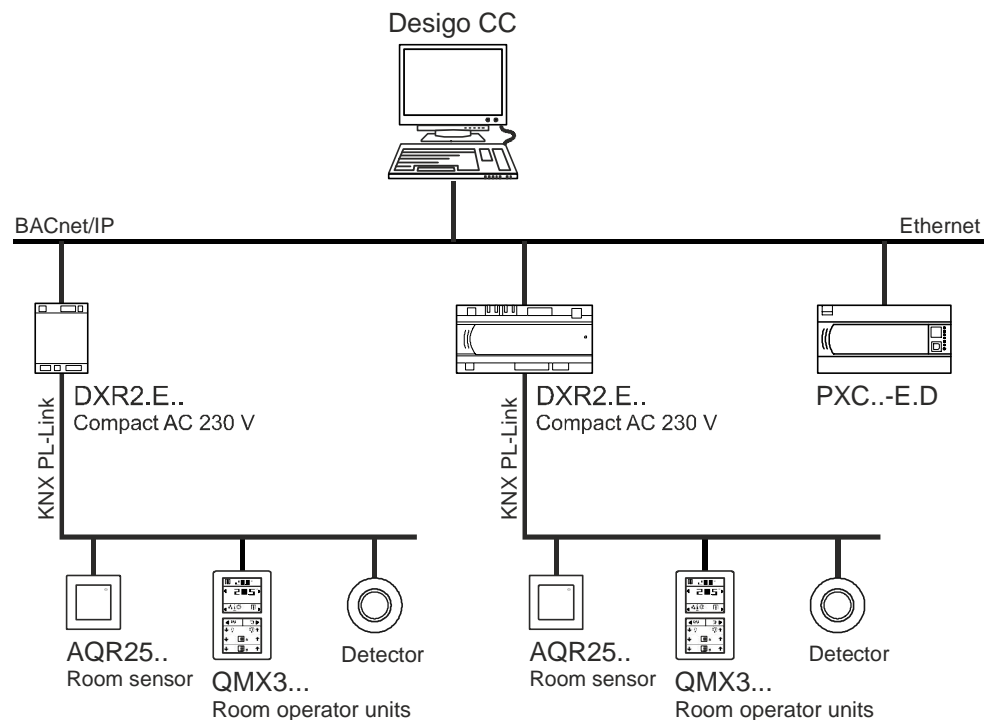
The stations communicate with each other and other system components, depending on the type, via BACnet/IP (DXR1.E../DXR2.E..) or BACnet MS/TP (DXR1.M../DXR2.M..). The room automation stations have a set number of I/O data points and an onboard interface to KNX to connect field devices. The automation stations are delivered with preloaded applications and only need to be configured.

A comprehensive library of proven, standardized applications is also available and can be used instead of the preloaded applications. Buttons, sensors, and actuators for lighting and shading are connected to the room automation stations via the KNX PL-Link.

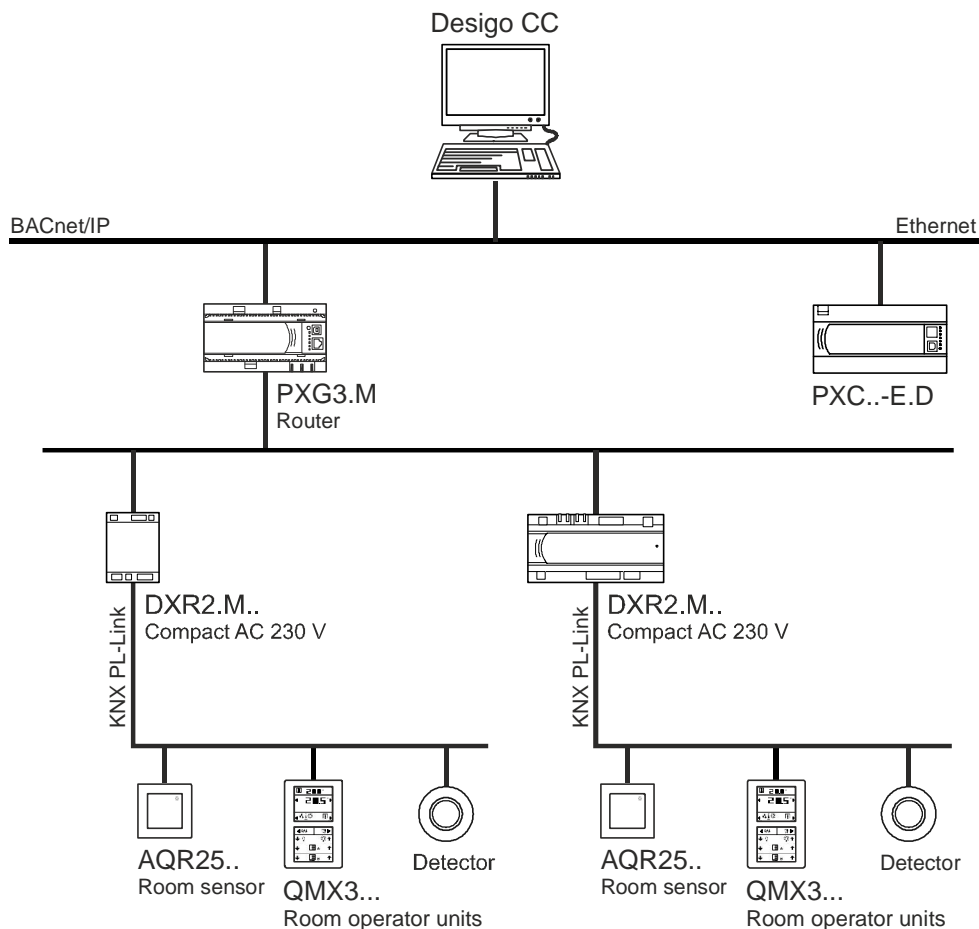
The preloaded and proven standardized applications in the library are configured using ABT Site and offer a great deal of flexibility since the inputs and outputs of the DXR1../DXR2.. can also be configured in addition to the functions.

See *Designo Configurable Room Automation (BACnet) – Product Range Description (A6V10866237)*.

Compact DXR2 room automation stations for BACnet/IP



Compact DXR2 room automation stations for BACnet MS/TP



Applications

The tables below show the functions of the different applications of the DXR2 room automation stations.

Preloaded applications

Application

Fan coil unit

- Outside Air Damper
- Single Speed Fan , Multi Speed Fan or Variable Speed Fan
- Chilled water cooling coil
- Direct expansion evaporator cooling coil
- Heating/Cooling coil
- Hot water heating coil
- Electric heating coil modulating, single stage or two stage
- Room temperature control by two-pipe system with change-over
- Room temperature control by four-pipe system
- Supply air temperature cascade control
- Room dehumidification control
- Air volume flow control
- Rapid ventilation
- Green leaf

Variable air volume

- Supply and extract air control
- External flow control for VAV with integrated flow controller and differential pressure sensor
- Internal flow controller and differential pressure sensor for damper actuator

Application

| | |
|------------------------------|---|
| | <ul style="list-style-type: none"> control • Internal flow controller and velocity sensor for damper actuator control • Chilled water cooling coil • Heating/Cooling coil • Hot water heating coil • Electric heating coil modulating, single stage or two stage • Room temperature control by two-pipe system with change-over • Room temperature control by four-pipe system • Supply air temperature cascade control • Air flow tracking for under/overpressure • Room dehumidification control • Room air quality control • Rapid ventilation • Green leaf |
| Radiator and chilled ceiling | <ul style="list-style-type: none"> • Chilled ceiling with chilled water • Heated/Chilled ceiling by two-pipe system with change-over • Heated/chilled ceiling by four-pipe system with 6 way valves • Heating ceiling with hot water • Hot water radiator • Electric radiator modulating or staged • DOWNDRAFT compensation for radiators • Condensation monitor • Room temperature control • Green leaf |
| Fan powered box | <ul style="list-style-type: none"> • Supply air control • External flow control for VAV with integrated flow controller and differential pressure sensor • Internal flow controller and differential pressure sensor for damper actuator control • Internal flow controller and velocity sensor for damper actuator control • Single Speed Fan , Multi Speed Fan or Variable Speed Fan • Chilled water cooling coil • Heating/Cooling coil • Hot water heating coil • Electric heating coil modulating, single stage or two stage • Room temperature control by two-pipe system with change-over • Room temperature control by four-pipe system • Supply air temperature cascade control • Room air quality control • Rapid ventilation • Green leaf |
| Four light groups | <ul style="list-style-type: none"> • Manual switched control • Manual dimmed control • Automatic presence control • Automatic brightness control • Constant light control |

Application

- Multi group constant light control
- LED support on push buttons
- Green Leaf - RoomOptiControl
- Burn-in & operating hours function

Two blinds

- Manual control
- Automatic control with anti glare function and energy efficiency function
- Green Leaf - RoomOptiControl
- Collision detection

Loadable central functions

Application

Central functions

- 4x Control room operating mode groups with:
 - Room mode and room setpoint distribution to rooms
 - Start optimization switches the heating on at the appropriate time
 - Three delayed distribution groups of room operating mode for big buildings
- 1x Seasonal compensation of room temperature setpoints
- 2x Demand controlled hot water supply group
- 2x Demand controlled chilled water supply group with:
 - Condensation prevention shifts the base chilled water setpoint to avoid condensation at chilled ceiling radiant devices
 - Two pipe changeover control handles the heating / cooling changeover for a two-pipe system
 - Free cooling manages the delivery of chilled water in situations where this can be done with almost a zero expenditure of energy, e.g., chiller plants with the possibility to cool the water with the recoolers when outside conditions are favorable.
- 1x Demand temperature control group with:
 - Relief function opens additional VAV without demand to ensure stable working of the primary plant
 - Changeover evaluation decide if the central air should be used for heating or for cooling
 - Dew point evaluation is used to dehumidify at the primary air handling unit to avoid condensation in the rooms
 - Humidity demand evaluates room humidity to help the primary plant determine when to humidify or dehumidify
 - Override function allows a technician or balancer to override the VAV applications for balancing and commissioning
- 1x Demand controlled pressure control by either:
 - Supply VAV position evaluation helps to optimize fan speed by averaging the 10 highest supply damper positions and providing this information to the central plant
 - Extract VAV position evaluation helps to optimize fan speed by averaging the 10 highest extract damper positions and providing this information to the central plant.
 - Supply VAV flow deviation helps to optimize the fan speed by calculating the airflow deviation through the supply VAV dampers
 - Extract VAV flow deviation helps to optimize the fan speed by calculating the airflow deviation through the extract VAV dampers
 - Supply VAV flow saturation evaluation (air flow control loop cannot get enough air to reach setpoint) evaluates the supply VAV saturation signals

Application

from the rooms to optimize the fan speed

- Extract VAV flow saturation evaluation (air flow control loop cannot get enough air to reach setpoint) evaluates the extract VAV saturation signals from the rooms to optimize the fan speed
- Supply VAV setpoint evaluation with the summed setpoints of the supply VAV the fan's speed can be set for an optimized fan speed when VAV positions and VAV flow rates are not known
- Extract VAV setpoint evaluation with the summed setpoints of the extract VAV the fan's speed can be set for an optimized fan speed when VAV positions and VAV flow rates are not known

Loadable central functions (continued)

Application

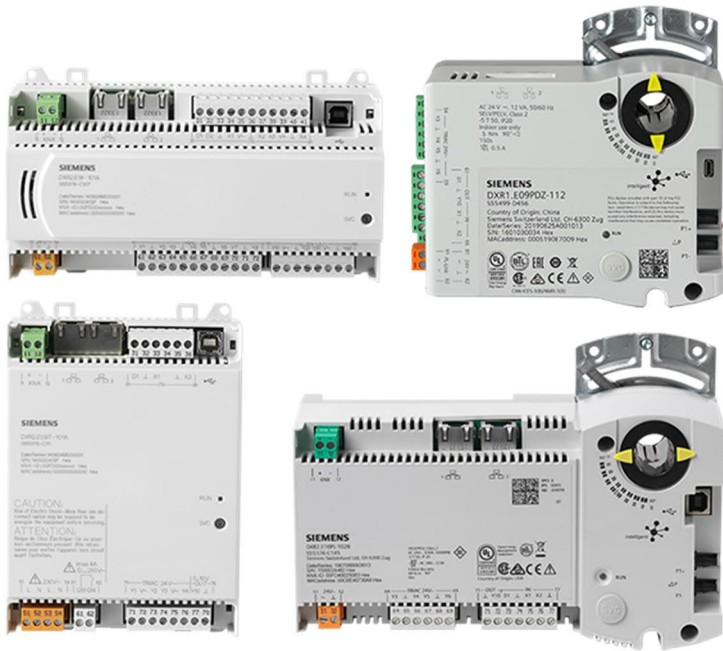
Central functions

- 2x VAV supply fire emergency group with off, extract, pressurization or purge
- 1x Central weather station with:
 - Outside temperature
 - Outside brightness
 - Outside solar radiation
 - Outside wind speed
 - Outside precipitation
- 2x Light manual central operation group
- 1x Light central control group for emergency situations
- 4x Shading central facade functions with:
 - Central weather station brightness calculation supports facade automatic function
 - Glare protection function calculates the glare protection state by central weather station for all facade
 - Annual shading calculates the glare protection state for all facade by information from annual shading computer
 - Thermal protection for unoccupied rooms by central global radiation sensor on weather station
 - Three delayed distribution groups for central blind commands for big buildings
- 2x Shading manual central operation with 3 delayed distribution groups for big buildings
- 1x Shading service ensures central commanding of blind group with high priority
- 1x Shading central protection for all blinds with:
 - Wind protection
 - Precipitation protection
 - Frost protection
 - Three delayed distribution groups for big buildings

See *Application Catalog*.

Compact room automation stations

DXR1 and DXR2 automation stations



DXR2 compact room automation stations

Communication

| | | | | | | | | | | |
|------------------------------|-----------------------|------------------------|-----------------------|-------------------------|--------------------------|-----------------------|------------------------|-------------------------|-----------------------|-----------------------|
| BACnet/IP | DXR2.E 09- 101A | DXR2.E 09T- 101A | DXR2.E 10- 101A | DXR2.E 10PL- 102B | DXR2.E 10PLX- 102B | | DXR2.E 12P- 102A | DXR2.E 12PX- 102B | DXR2.E 18- 101A | DXR2.E 18- 102A |
| BACnet MS/TP ¹ | DXR2. M09- 101A | DXR2. M09T- 101A | DXR2. M10- 101A | DXR2. M10PL- 102B | DXR2. M10PL X-102B | DXR2. M11- 101A | DXR2. M12P- 102A | DXR2. M12PX- 102B | DXR2. M18- 101A | DXR2. M18- 102A |

Applications

| | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| Room operating | • | • | • | • | • | • | • | • | • | • |
| Heated / Chilled ceiling radiator | • | • | • | • | • | • | • | • | • | • |
| Fan coil unit | • | • | • | | | • | | | • | |
| VAV system | | | | • | • | | • | • | | • |
| Lighting | • | • | • | • | • | • | • | • | • | • |
| Shading | • | • | • | • | • | • | • | • | • | • |
| Central functions ¹ | | | | | | | | | • | • |

Housing

| | | | | | | | | | | |
|------|---|---|---|----------------|----------------|---|---|---|---|---|
| DIN | | | | | | • | • | • | • | • |
| Flat | • | • | • | • ² | • ² | | | | | |

Operating voltage

| | | | | | | | | | | |
|------|---|---|---|---|---|---|---|---|---|---|
| 230V | • | • | • | | | | | | | |
| 24V | | | | • | • | • | • | • | • | • |

Inputs and outputs onboard

| | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| Digital inputs | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 2 | 2 |
| Universal inputs | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 4 | 4 |
| Relay outputs | 3 | 1 | 3 | | | | | | | |
| Triac outputs | | 4 | 4 | 4 | 4 | 6 | 6 | 6 | 8 | 8 |
| Analog outputs (DC 0...10 V) ³ | 3 | 1 | | 1 | 1 | 2 | 2 | 2 | 4 | 4 |
| Pressure sensor | | | | 1 | 1 | | 1 | 1 | | |

Maximum configuration

| | | | | | | | | | | |
|--|----|----|----|----|----|----|----|----|----|----|
| Number of I/O data points ⁴ | 30 | 30 | 30 | 30 | 60 | 30 | 30 | 60 | 60 | 60 |
| Integrated power supply for KNX (mA) | 50 | 50 | 50 | 50 | 50 | 50 | 50 | 50 | 50 | 50 |

Key

- ¹ Cannot be combined with other applications.
- ² Mounting via damper shaft.
- ³ Cannot be extended by KNX PL-Link inputs and outputs.
- ⁴ Total number of data point used by TX-I/O, KNX PL-Link and DALI. For details, see chapter *System Configuration*.

See *Compact room automation stations, BACnet/IP, 230 V DXR2.E10.., DXR2.E09.., DXR2.E09T..* (N9204).

See *Compact room automation stations, BACnet/IP, 24 V DXR2.E18.., DXR2.E12P..* (N9205).

See *Compact room automation stations, BACnet MS/TP, 230 V DXR2.M10.., DXR2.M09.., DXR2.M09T..* (N9206).

See *Compact room automation stations, BACnet MS/TP, 24 V DXR2.M11.., DXR2.M12P.., DXR2.M18..* (N9207).

DXR1 compact room automation stations

Communication

| | | | | | |
|---------------------------|------------------|------------------|----------------|----------------|------------------|
| BACnet/IP | DXR1.E09PD Z-112 | DXR1.E09PD Z-113 | DXR1.E10PL-112 | DXR1.E10PL-113 | DXR1.E02PL Z-112 |
| BACnet MS/TP ¹ | DXR1.M09PD Z-112 | DXR1.M09PD Z-113 | | | |

Applications

| | | | | |
|--------------------------|---|---|---|---|
| Room operating | • | • | • | • |
| Heated / Chilled ceiling | | | • | • |

| | | | | | |
|---|----------------|----------------|----------------|----------------|----------------|
| radiator | | | | | |
| VAV system | • | • | • | • | • |
| Central functions ¹ | | | | | |
| Housing | | | | | |
| DIN | | | | | |
| Flat | • ² | • ² | • ² | • ² | • ² |
| Operating voltage | | | | | |
| 230V | | | | | |
| 24V | • | • | • | • | • |
| Inputs and outputs onboard | | | | | |
| Digital inputs | | | 1 | 1 | |
| Universal inputs | 2 | 2 | 2 | 2 | |
| Relay outputs | | | | | |
| Triac outputs | 4 | 4 | 4 | 4 | |
| Analog outputs (DC 0...10 V) ³ | 1 | 1 | 1 | 1 | |
| Pressure sensor | 1 | 1 | 1 | 1 | 1 |
| Maximum configuration | | | | | |
| Number of I/O data points ⁴ | 16 | 16 | 22 | 19 | 2 |
| Integrated power supply for KNX (mA) | | | 50 | 50 | |

Key

- ¹ Cannot be combined with other applications.
- ² Mounting via damper shaft.
- ³ Cannot be extended by KNX PL-Link inputs and outputs.
- ⁴ Total number of data point used by TX-I/O, KNX PL-Link and DALI. For details, see chapter *System Configuration*.

Accessories for DXR1 automation stations

Accessory device

| | |
|-----------|---|
| QMA1.N30H | Room sensor for DXR1 room automation stations |
| QMX1.M34H | Room unit for DXR1 room automation stations |
| DXA.T50 | Air tube accessory kit 50 mm |

See *Compact room automation station, BACnet/IP, AC 24 V (Actuating DXR) DXR1.E02PLZ-112 (A6V11393935)*

See *Compact actuating room automation stations, BACnet/IP, AC 24 V (Actuating DXR) DXR1.E09PDZ-112, DXR1.E09PDZ-113 (A6V11393931)*

See *Compact actuating room automation stations, BACnet/IP, AC 24 V (Actuating DXR) DXR1.E10PL-112, DXR1.E10PL-113 (A6V11393933)*

See *Compact actuating room automation stations, BACnet MA/TP, AC 24 V (Actuating DXR) DXR1.M09PDZ-112, DXR1.M09PDZ-113 (A6V11393929)*

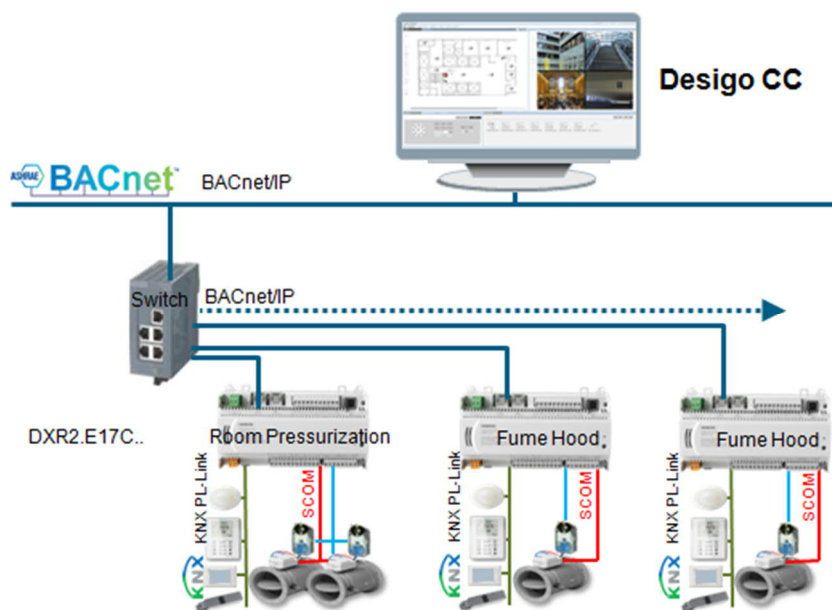
Applications for DXR1

| Application | Functions |
|---------------------|---|
| Variable air volume | <ul style="list-style-type: none"> • Supply and extract VAV with integrated damper actuator and heating/chilled ceiling • Room temperature control • Air quality control • Relative humidity monitor • Room temperature and rapid ventilation operation via KNX PL-Link room operator unit with temperature, air quality & relative humidity measurement • Supply VAV with integrated damper actuator and staged electric reheater • Supply air volume control with integrated damper actuator • Air volume control with integrated damper actuator |
| Fan powered box | <ul style="list-style-type: none"> • Fan powered box with series fan and staged electric reheater • Supply air volume control with integrated damper actuator (5 Nm / 10 Nm) • Room temperature control • Air quality control • Relative humidity monitor • Fan powered box with variable speed fan and staged electric reheater • Room temperature and rapid ventilation operation via KNX PL-Link room operator unit with temperature, air quality & relative humidity measurement |

Room pressurization and fume hood control

Designo offers a range of air volume flow components and accessories for secure, precise and fast measurement, control and monitoring of air volume flows and room pressures in highly specialized working spaces.

See *Designo Configurable Room Automation (BACnet) – Product Range Description (A6V10866237)*.



Components for room pressurization and fume hood control

Component

DXR2.E17C.. Room automation station, BACnet/IP, 24VAC, 17 I/Os, 30 data points

Component

| | |
|---------------|--|
| DXR2.E17CX.. | Room automation station, BACnet/IP, 24VAC, 17 I/Os, 60 data points |
| QMX3.P87-1WSC | Operating display panel, wall mounted (KNX) |
| QMX3.P88-1WSC | Operating display panel, thin and flush mounted (KNX) |
| n/a | Room condition monitor (BACnet/IP) |

Accessories for room pressurization and fume hood control

Accessory device

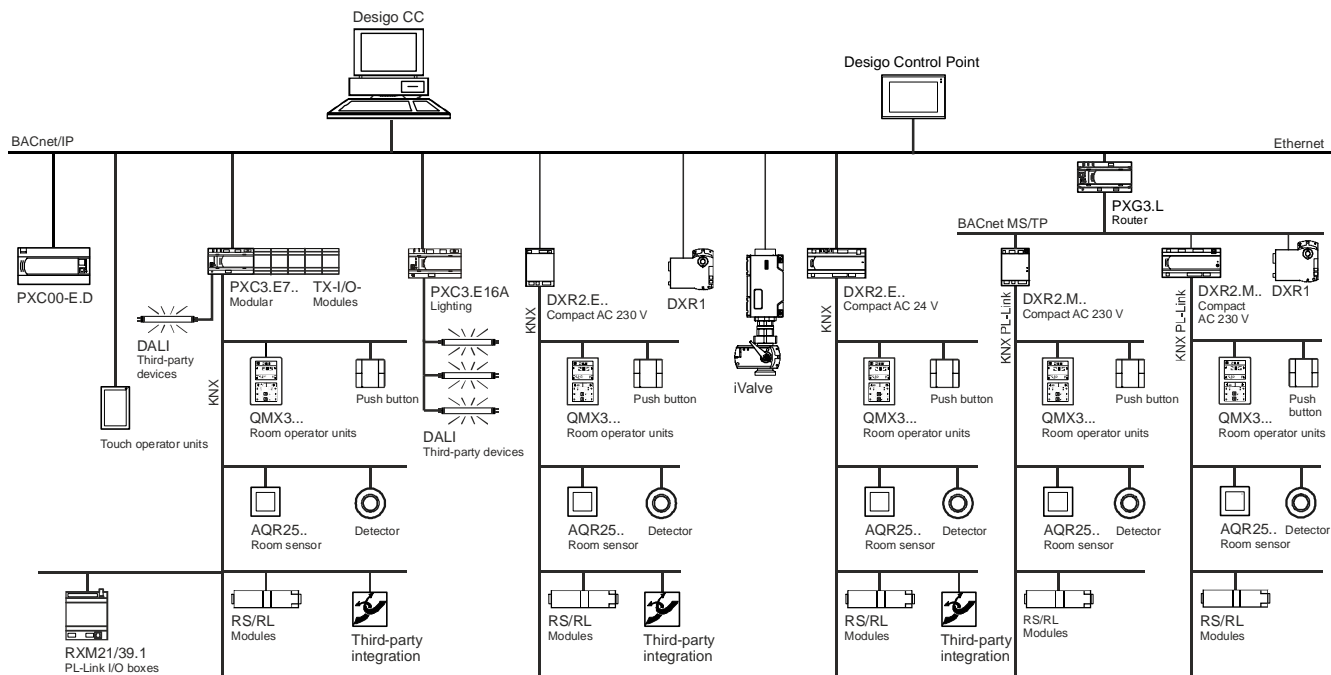
| | |
|-------------|---|
| n/a | Sash wire sensor |
| DXA.S04P1 | Air flow pressure sensor (SCOM) |
| DXA.S04P1-B | Air flow pressure sensor with IP65 box (SCOM) |
| DXA.S12C | Sash open area module (SCOM) |

20.1.2 Programmable

The DXR2.. and PXC3.. room automation stations are programmable, based on proven application blocks. Thus, solutions can be tailored to specific needs and can achieve maximum efficiency and comfort.

See Range Description *Designo Room Automation (BACnet)*, *Programmable Room Automation - Emergency Lighting (A6V10640596)*, *Programmable Room Automation - Room Operation (A6V10640597)*, *Programmable Room Automation - Distributed Functions and Scenes (A6V10640598)* and *Programmable Room Automation - Lighting Controls and DALI (A6V10640599)*.

Topology



Applications

A comprehensive block library for room automation is provided as part the scope of delivery. The library contains predefined application functions for room climate, lighting, shading, and superimposed room functions. The applications can be combined with operating and display functions as required. The individual application

functions can be adapted to customer needs and are programmable. The application functions do not depend on the selected field devices.

See *Application Catalog*.

Configuration of application functions

Many application functions are preconfigured and available in the library. Retroactive configuration during engineering or commissioning is possible. Your own configured application functions and entire rooms can be stored in a project library.

Configuration of field devices

The application architecture does not depend on the field device interface. Field devices can be connected directly to the PXC3 room automation station (via TX-I/O modules) or via bus (KNX or DALI) or IP communication.

Many field devices are preconfigured and available in the library. Retroactive configuration during engineering or commissioning is possible. Project-specific field devices configured accordingly can be saved in a project library.

Modular room automation stations

PXC3 room automation stations with control functions for one or multiple rooms:

- Assume control functions for one or multiple rooms.
- Communicate with each other or other system components via BACnet/IP. Scope and functionality of supported BACnet objects are matched to the requirements of room automation.
- Provide a 2-port Ethernet interface for cost-effective cabling via daisy chaining.
- Contain bus supplies for island bus, KNX PL-Link, and DALI. Internal bus supplies can be extended for island bus and KNX PL-Link as needed.
- Have an integrated web server for IP communication with touch room operator units.

See *Room automation station PXC3.E7..* (CM1N9203).



| | PXC3.E72 | PXC3.E72A | PXC3.E75 | PXC3.E75A | PXC3.E16A |
|---|-----------|-----------|-----------|-----------|-----------|
| Typical number of rooms / room segments | 4/8 | 4/8 | 8/16 | 8/16 | n/a |
| System communication | BACnet/IP | BACnet/IP | BACnet/IP | BACnet/IP | BACnet/IP |
| HMI automation level | | | | | |
| QMX3 | • | • | • | • | |
| QMX7 | • | • | • | • | • |
| Web based test and setup tool | • | • | • | • | • |
| System functions (BACnet) | | | | | |
| BACnet profiles | B-ASC | B-ASC | B-ASC | B-ASC | B-ASC |
| Programming | • | • | • | • | • |
| Peripheral bus | | | | | |
| Bus for I/O module | • | • | • | • | |
| KNX PL-Link ¹ / KNX S-Mode | • | • | • | • | |
| DALI | | • | | • | • |
| Maximum configuration | | | | | |
| Number of I/O data points ² | 140 | 140 | 280 | 280 | 64 |

| | PXC3.E72 | PXC3.E72A | PXC3.E75 | PXC3.E75A | PXC3.E16A |
|--------------------------------------|----------|-----------|----------|-----------|-----------|
| Inputs/Outputs for TX I/O modules | 72 | 72 | 200 | 200 | 0 |
| Devices on KNX PL-Link | 64 | 64 | 64 | 64 | 0 |
| DALI ballasts | | 64 | | 64 | 64 |
| Integrated power supply for KNX (mA) | 160 | 160 | 160 | 160 | n/a |

Key

- ¹ Dedicated devices with KNX PL-Link.
² Total number of data points used by TX-I/O, KNX PL-Link and DALI. For details, see chapter *System Configuration*.

TX-I/O modules

TX-I/O modules (TXM1) help connect field devices to the PXC3 room automation station. Bus supply and interface modules (TXS1, TXA1) are available as an accessory.



The following TX-I/O modules can be used with the PXC3 room automation station:

- TXM1.8T: Triac module with 8 outputs (AC 24 V) to control thermal and motorized valve actuators (AC 24 V) for up to 4 actuators (3-point output) or 8 actuators (permanent contact or pulse width modulation).
- TXM1.6RL: Bistable relay module to switch lighting for up to 6 data points.
- TXM1.8RB: Relay module to control blind motors for up to 2 motors (3 end switches) or 4 motors (2 end switches).
- TXM1.16D: Digital input modules for up to 16 data points.
- TXM1.8D: Digital input modules for up to 8 data points.
- TXM1.6R: Relay module for up to 6 data points.
- TXM1.8U: Universal module for up to 8 data points.

See *TX-I/O Assortment overview (CM2N8170)*.

Compact room automation stations

| Communication | | | | | | |
|----------------------------|-------------------|--------------------|-------------------|-------------------|---------------|-------------------|
| BACnet/IP | DXR2.E09 -101A | DXR2.E09 T-101A | DXR2.E10 -101A | | DXR2.E12 P | DXR2.E18 -1..A |
| BACnet MS/TP | DXR2.M09 -101A | DXR2.M09 T-101A | DXR2.M10 -101A | DXR2.M11 -101A | DXR2.M12 P | DXR2.M18 -1..A |
| Housing | | | | | | |
| DIN | | | | • | • | • |
| Flat | • | • | • | | | |
| Operating voltage | | | | | | |
| 230V | • | • | • | | | |
| 24V | | | | • | • | • |
| Inputs and outputs onboard | | | | | | |
| Digital inputs | 1 | 1 | 1 | 1 | 1 | 2 |
| Universal inputs | 2 | 2 | 2 | 2 | 2 | 4 |
| Relay outputs | 3 | 1 | 3 | | | |
| Triac outputs | | 4 | 4 | 6 | 6 | 8 |

| | | | | | | |
|--------------------------------------|----|----|----|----|----|----|
| Analog outputs (DC 0...10 V)* | 3 | 1 | | 2 | 2 | 4 |
| Pressure sensor | | | | | 1 | |
| Maximum configuration | | | | | | |
| Number of I/O data points | 30 | 30 | 30 | 30 | 30 | 60 |
| Integrated power supply for KNX (mA) | 50 | 50 | 50 | 50 | 50 | 50 |

Key

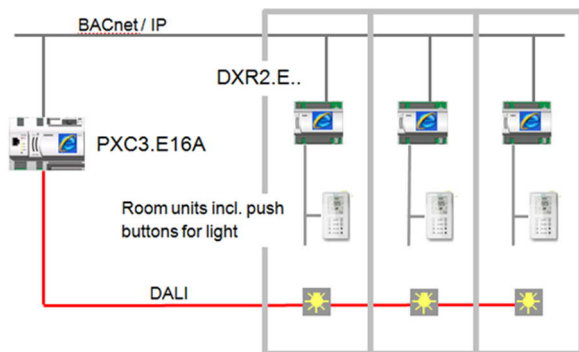
¹ Total number of data points used by TX-I/O, KNX PL-Link and DALI. For details, see chapter *System Configuration*.

PXC3.E16A room automation station for lighting

The PXC3.E16A room automation station is tailored for challenging lighting applications. All lighting applications that also run on the PXC3.E7.. are available. The PXC3.E16A communicates via BACnet/IP with the DXR2.E.. and PXC3.E.. room automation stations. Using the on-board DALI interface, up to 64 ECGs (electronic control gear) can be integrated in 16 groups. The PXC3.E16A can be used for centralized lighting automations or, as applicable, as a supplement to a decentralized HVAC installation.

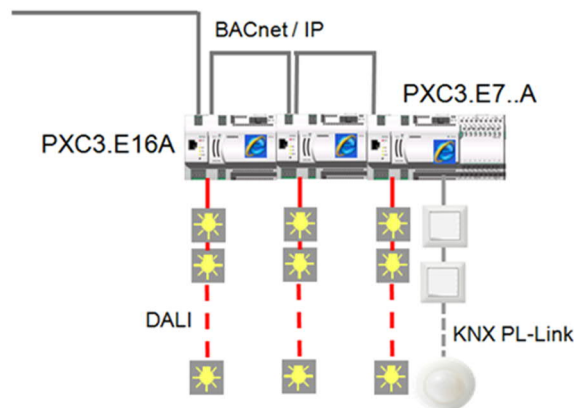
Example: Centralized lighting installation with decentralized HVAC installation

- DXR2.E.. to automate HVAC in the room
- Centralized installation with PXC3.E..A for lighting



Example: Centralized lighting installation without HVAC installation

- One PXC3.E16A is centrally installed per DALI line
- Optional PXC3.E7..A
 - To integrate buttons via KNX PL-Link
 - To use TXM1 modules
 - Three-phase power installation possible

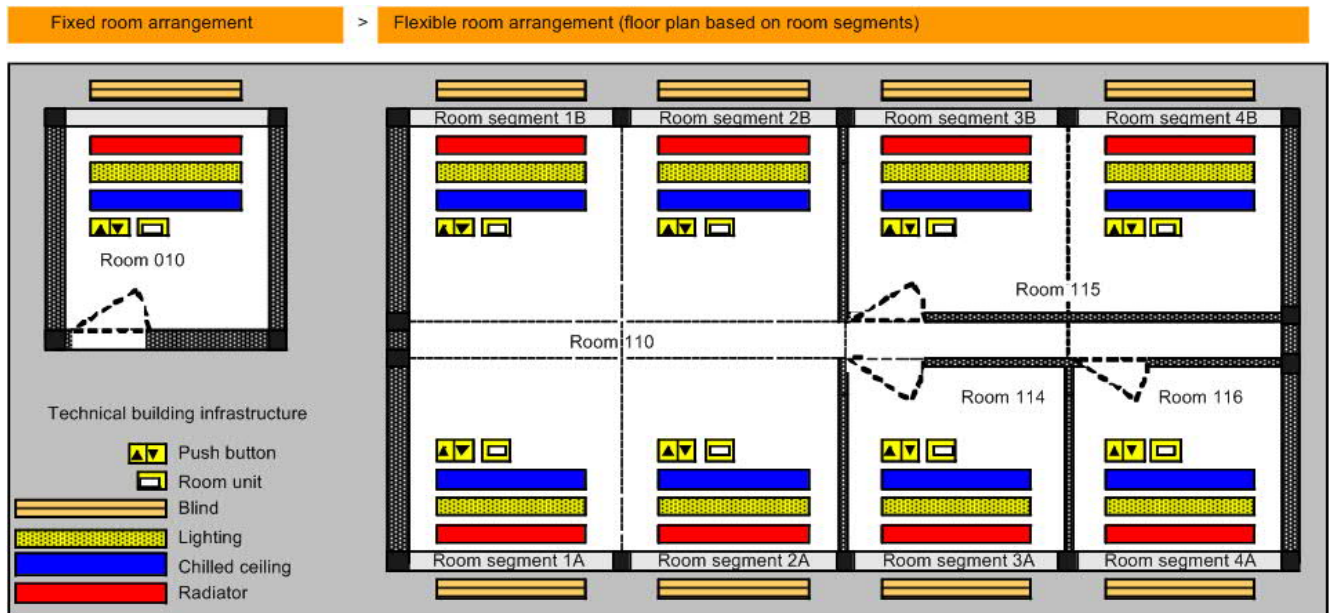


20.1.3 Rooms and room segments

There are two methods to structure a building:

- Rooms (with fixed walls)
- Room segments (typically based on movable walls)

One of the two methods or a mixture thereof is possible depending on the building structure or required flexibility (e.g., during the usage phase).



A room segment is the smallest indivisible element. A room comprises at least one or several adjacent room segments. A room segment is defined and created only once. Room segments are typically combined several times to rooms over the course of a building's lifecycle.

20.1.4 Central control functions and grouping

Grouping is used to implement central control functions and to coordinate demand and forced signals from the various rooms in an entire building, building wing, floor, etc.

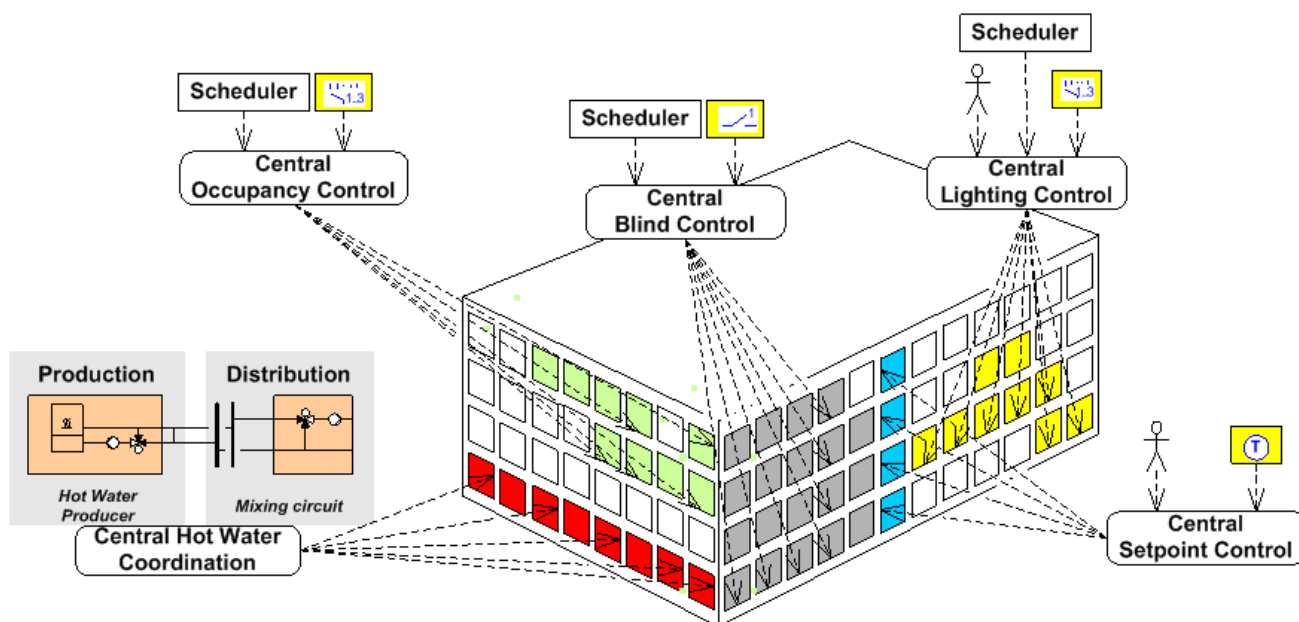
Hidden behind the central control functions are system functions, such as operator interventions via BACnet clients, schedulers, automatic reactions, and data from a weather station.

Central functions influence:

- Room operating mode (occupancy and use in room)
- HVAC control via various setpoint requirements depending on the room operating mode
- HVAC setpoints via a weather-dependent adjustment
- Lighting control
- Shading control (blinds)

Grouping can be used to coordinate demand, operating, and forced signals, that is:

- Request signals for hot water distribution (heating circuit)
- Request signals for chilled water distribution (cooling circuit)
- Record demand, operating, and forced signals for the primary air handling unit



Various sources are available for forming these central superposed functions:

- External system or third-party device
- System user via BACnet client
- Building user via BACnet client or local operator unit
- Scheduler or reaction program
- Superposed office based on grouping function

They are distributing after evaluating signals and commands via a Grouping function.

One group master exists for each of the various categories which then forwards the resulting information to all assigned group member (rooms). A group master can for its part be a group member of a superposed group master.

20.1.5 Designo room automation and the management level

See chapter *Designo room automation integration*.

20.1.6 Designo room automation and the automation level

Alarming is triggered directly on the PXC3.E.. and DXR2.. room automation stations. A primary automation station (typically PXC00.E-D) is only used for calendars, schedulers and time setting. This simplifies engineering and reduces the number of required system components.

20.2 Designo RXB

The Designo RXB room automation system controls and monitors comfort conditions in individual rooms. It supplies predefined solutions for HVAC.

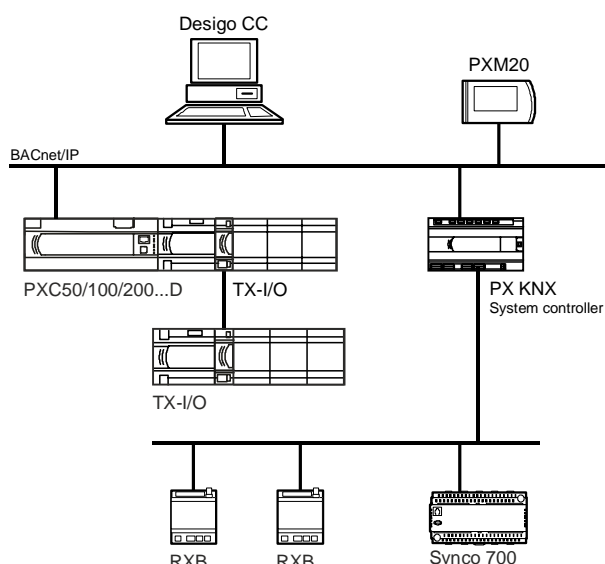
See *RXB Room automation system - system overview* (CM110380).

The range consists of controllers, operator units and predefined applications. The applications are configured and downloaded with the ETS Professional commissioning and service tool.

See *Working with ETS* (CM1Y9779).

RXB topology

The Designo RXB room automation system is based on KNX/EIB technology. To integrate Designo RXB into the automation level, the RXB data is mapped to BACnet.



Group address / Binding

A group address / binding is a connection of network variables of the same type between different nodes. The group addresses / bindings are generated using ETS (EIB tool software) when designing the KNX/EIB network. The bound network variables communicate when changing the value and using a heartbeat. Transmission and reception times are also monitored, allowing you to react to communications errors.

Discipline I/Os

Discipline I/Os are function blocks in the PX KNX system controller, which gather data from the RXB controller and make it available on the BACnet network. Discipline I/Os are available for HVAC functions.

Konnex Association

The Konnex Association is an association founded by the manufacturers of KNX/EIB products, to define interoperability guidelines for KNX/EIB systems. The association is responsible for checking compliance and for the certification of KNX/EIB products.

KNX/EIB nodes

A KNX/EIB node is a device which is connected to the KNX/EIB bus and communicates with other KNX/EIB nodes.

Network variables (NV)

Network variables (NV) allow the exchange of data between different KNX/EIB nodes. Network variables may be input or output variables.

Room-based groups

The discipline I/Os representing the RXB controllers in a room are combined in the PX KNX system controller into a room-based group. The result is a room view.

Cross-room groupings

Cross-room groupings contain all the control variables common to a given user grouping (e.g., north facade, tenant A, west zone, etc.) and distribute these control variables to the associated room or group members.

PX KNX system controller

The PX KNX system controller comprises a PXC001(-E).D controller and loaded PX KNX firmware. Communication takes place via BACnet/LonTalk (PXC001.D) or BACnet/IP (PXC001-E.D). With the system controller, you can integrate the Synco RMU710, RMU720, RMU730 and RMH760 controllers into Desigo.

PX KNX Tool

The PX KNX tool is used to configure the PX KNX system controller on the KNX side.

20.2.1 Product range overview

Desigo RXB is an innovative range of controllers and room units. Data communication is based on KNX/EIB technology.

Desigo RXB hardware

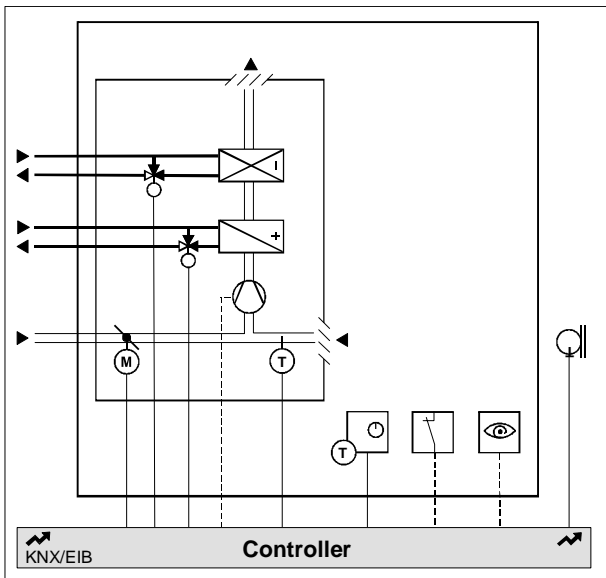
The range comprises compact controllers, easy-to-operate room units and controllers in room-style housings. The input and output configurations of the controllers, and the housing style are fully optimized to suit their field of application. The HVAC functions are operated with standard room units or controllers in a room-style housing.

Desigo RXB software

Each controller is loaded with a selection of application software which contains the control program for the associated room or area within a room.

The ETS commissioning and service tool is used for the engineering and commissioning of a network incorporating the Desigo RXB range. This tool also supports the creation of communication bindings between KNX/EIB-compatible devices (Desigo RXB or third-party devices).

Example: Fan coil system



| Application | Name | Controller |
|-------------|---|---------------|
| FNC02 | Two-pipe system with changeover | RXB21.1/FC-10 |
| FNC04 | Four-pipe system | RXB39.1/FC-13 |
| FNC08 | Four-pipe system with room supply air cascade control | |
| FNC20 | Four-pipe system with damper control | |
| FNC03 | Two-pipe system with changeover and electric reheater | RXB22.1/FC-12 |
| FNC05 | Four-pipe system with electric reheater | RXB39.1/FC-13 |
| FNC10 | Two-pipe system with changeover and outside air | RXB21.1/FC-11 |
| FNC12 | Four-pipe system with outside air | |
| FNC18 | Two-pipe system with changeover and radiator | |

Common functions:

- Window contact, occupancy sensor, four operating modes
- Manual fan control with room unit
- Automatic fan control (three speeds)
- Options with two-pipe systems: Heating only, cooling or changeover via KNX/EIB bus

20.2.2 RXB and the management level

Generic and engineered operation is available at the management level.

See *Desigo CC System Description* (A6V10415500)

20.2.3 RXB and the automation level

Desigo RXB is integrated into the automation level with the PX KNX system controller.

The main tasks of the system controller are:

- Mapping RXB data to BACnet objects
- Implementing higher-level functions (grouping, time schedules, etc.)

On the BACnet side of the PX KNX system controller, the RXB controllers can be operated and monitored from a client. Data can also be exchanged with the primary plant.

20.2.4 RXB applications

The existing Desigo RXB applications cannot be modified for a specific project by the user. These applications are preprogrammed by groups in the controller and are selected and parameterized using the ETS Professional commissioning and service tool.

Applications of the same type are grouped into application groups. The technical manual contains the complete range of RXB applications.

See *RXB (KNX) Technical manual* (CM110389).

RXB application library

The RXB application library contains application groups, each of which contains applications of the same type. The RXB application library has a version number which is defined in the RXB Valid Version Set. This Valid Version Set also defines the version of each individual RXB application.

Application groups

Similar application types are grouped into application groups. These differ from each other in terms of how the functions are implemented. Thus chilled ceiling with radiator (CLC02) and chilled ceiling and electric radiator (CLC03) are two different applications within the CLC group. The first of these two applications uses water for heating, while the second uses electrical energy. The difference between applications in the other groups follows a similar pattern.

The following application groups are available for RXB:

- CLC Chilled ceiling applications (not for Synco)
- FNC Fancoil applications
- VAV Variable air volume applications (not for Synco)

Individual applications

The individual application is designed for typical HVAC systems as commonly used in practice in individual rooms.

Configuring applications

Each application has a defined number of configuration parameters with which the application can be programmed for a specific project. These parameters consist both of general values (e.g., temperature setpoints, etc.) and of specific values for the application concerned (e.g., changeover configuration, electric reheater, etc.).

20.2.5 Mapping RXB in the PX KNX system controller

The RXB system is mapped to the PX KNX system controller with objects. These objects are called discipline I/Os and are components of the block library.

See *PX KNX, RXB integration – S-Mode* (CM1Y9775).

The following types are available for RXB:

- HVAC: Comprises all the HVAC information
- Shared: Contains shared data points (e.g., time schedules, occupancy status, etc.)

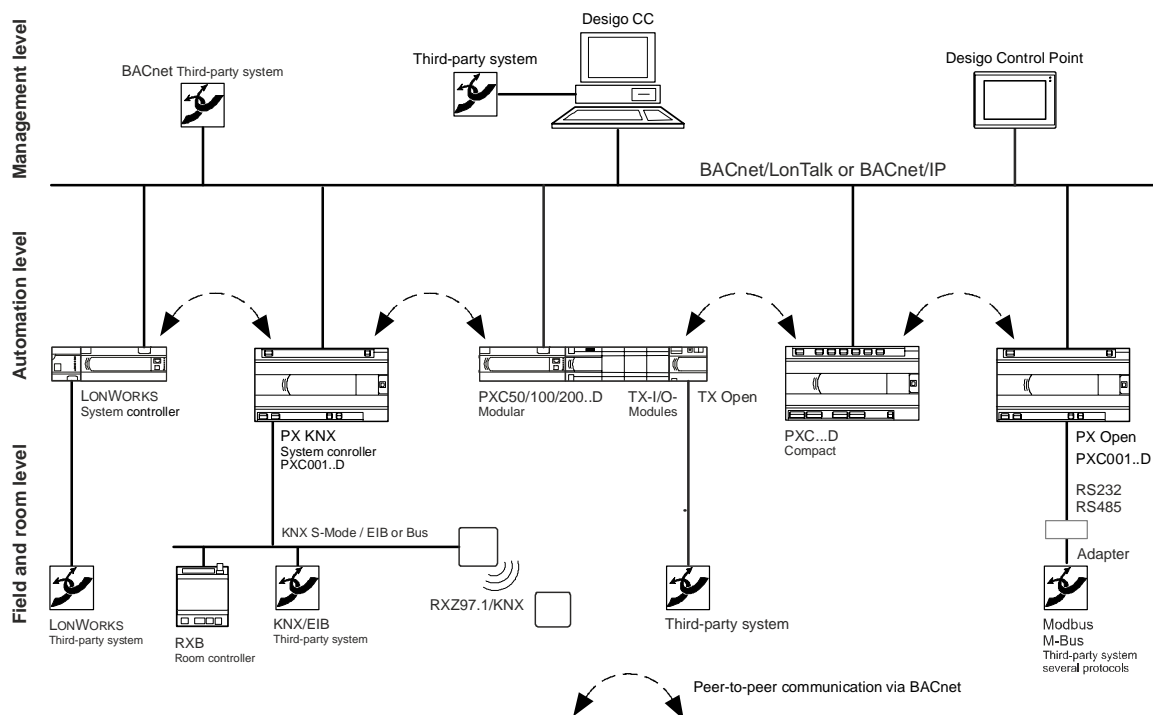
21 Desigo Open

Desigo Open lets you integrate devices and systems from different manufacturers into the Desigo system. Integration with Desigo Open offers:

- Standardized automated functions, operating and monitoring of the entire building
- Single-station operation, common view and display. Simplified multidisciplinary operation, common reporting and common alarm management.
- Peer-to-peer interaction, communication on the automation level, automated interactions and data exchange
- Comfort combined with lower energy consumption. New opportunities to save energy with systems that communicate among themselves. Improved performance, efficiency evaluation, flexibility and ability to modify system operation and configuration without re-cabling or new hardware.
- Engineering of integrated solutions in Xworks Plus (XWP)
- Reduced risk thanks to standard solutions. Clear functions that cover the most important standard protocols.

Topology

Third-party devices and systems can be integrated with Desigo on all levels.



Which protocols does Desigo support?

| Desigo Open System | Protocol |
|--|---|
| Desigo CC | SCADA, etc. |
| PX Open | Modbus, KNX/EIB, LonWorks, M-Bus, SCL, etc. |
| TX Open | Modbus, M-Bus, GENIbus, etc. |
| Room level (Desigo room automation and RX) | DALI, KNX, EnOcean, LonWorks |

Which plant sections can be integrated into Desigo on which level?

| Desigo Open system | Desigo Open application | Data points |
|-----------------------------|--|----------------|
| Desigo management platform | Desigo CC Energy monitoring, fire security, access control and security | 1,000 - 10,000 |
| Desigo PX | PX Open Power distribution, refrigeration machines | 50 - 2,000 |
| Desigo TX-I/O | TX Open Pumps, variable speed drives, meters, etc. | Max. 160 |
| Desigo room automation / RX | PXC3 Lighting and blinds | 16 DALI groups |

SDKs

If a solution is not supported by HQ and RCs need a specific solution, HQ offers Software Development Kits (SDKs) for experts. The regional companies can develop their own solutions using Software Development Kits (SDKs).

The following SDKs are available:

- PX Open platform SDK
- TX Open platform SDK

21.1 Integration on management level

The integration of third-party devices and systems on the management level is appropriate:

- For monitoring and operating plants that are not time-critical
- When process communication with other automation stations is not needed

Desigo CC

BACnet

BACnet is a widely used communication protocol for building automation and control networks. It defines a number of objects, services and data link layers. It is an essential part of Desigo CC's openness for integrating any third-party devices, using the BACnet/IP protocol. An online auto-discovery and alternatively an offline EDE import are available for integrating third-party devices.

See *BACnet 3rd party Integration Guide* (A6V10446271).

BTL

For more information about compliance and interoperability of Desigo products, go to <http://www.bacnetinternational.net/btl> and search for Desigo.

Modbus-TCP

A native Modbus-TCP driver lets you integrate a Modbus TCP server and subsequent Modbus RTU devices via a protocol converter. An offline importer supports the engineering workflow for integrating Modbus data points.

See *Modbus Integration Guide* (A6V10438039).

OPC

OLE for Process Control (OPC) is a communication standard for exchanging data between windows based software applications and process control hardware without any proprietary restrictions. It is a client/server technology, where one application acts as the server providing data, and another acts as a client using data. The most common specification *Data Access* (DA) defines a set of objects, interfaces and method to facilitate the interoperability. OPC has been extended to become a cross-platform communication standard, named OPC Unified Architecture (OPC UA).

For more information about OPC, see the documentation by the OPC Foundation (www.opcfoundation.org) and the OPC Training Institute (www.opcti.com).

OPC DA client

An OPC client interface lets you integrate any OPC server, using the Data Access specification. An offline Importer supports the engineering workflow to integrate OPC items.

See *OPC Server Integration Guide* (A6V10415483).

OPC DA server

An OPC server option provides a freely configurable set of data points for integration in any enterprise system, using the OPC DA standard. Each data point (object) is represented by several OPC items, providing the relevant readable and writable object property information.

See *OPC DA Server Manual* (A6V10415485).

The Desigo CC OPC server is officially tested and certified by the OPC foundation (<https://opcfoundation.org/products/view/251>).

OPC UA server

OPC Unified Architecture (OPC UA) clients can connect to the Desigo CC OPC DA server using the OPC DA/UA wrapper, provided with the Desigo CC setup. The UA wrapper meets the security model of mutual authentication for a trusted connection between the OPC UA server and the OPC UA client.

See *OPC DA Server Manual* (A6V10415485).

SNMP

Simple Network Management Protocol (SNMP) is a data communication protocol for monitoring devices and applications on a network. It is an Ethernet based protocol for retrieving management data from networked devices, and exposing this data as properties.

SNMP gives you the capability to monitor a device, e.g., a printer or UPS, which is not directly configured on a computer, but can be reached through a network link.

Device monitoring capabilities are provided by device manufacturers via a Management Information Base (MIB) text file, which describes the structure of the device management data. MIB files use a hierarchical namespace containing object identifiers (OID). Each OID identifies a property that can be read or written via SNMP.

Desigo CC has an SNMP Manager feature for reading and writing information from SNMP agents.

See *SNMP Application Guide* (A6V10455382).

Web services

Using RESTful technology, Desigo CC provides alarm, object and time series data via web based services for supervising management platforms or other third-party external applications.

21.2 Integration on automation level

The integration of third-party devices and systems on the automation level is appropriate when:

- Cross-communication to other PX or BACnet devices is needed
- System functions (e.g., alarms, trends, schedulers) are needed

The PX Open Platform comprises:

- PXC001.D system controller for the integration of KNX, Modbus, M-Bus and SCL via BACnet/LonTalk
- PXC001-E.D system controller for the integration of KNX, Modbus, M-Bus and SCL via BACnet/IP
- PXA40-RS1 and PXA40-RS2 option modules for additional data points

The automation stations have interfaces to RS232, RS485 and KNX.

Xworks Plus (XWP) is used to engineer all solutions. Various compounds and blocks are available.

The following solutions on the PX Open platform are available:

- PX KNX
- PX Modbus
- PX M-Bus
- PX SCL
- PX RS-Bus
- PX Pronto
- PX Open Platform (SDK)

Data points

| | PXC001.D | PXC001-E.D | PXA40-RS1 | PXA40-RS2 |
|-----------|----------|------------|-----------|-----------|
| PX KNX | 2,000 | 2,000 | n/a | n/a |
| PX Modbus | 250 | 250 | 800 | 2,000 |
| PX M-Bus | 250 | 250 | 800 | 2,000 |
| PX SCL | 250 | 250 | 800 | 1,000 |
| PX RS-Bus | 2,000 | 2,000 | n/a | n/a |
| PX Pronto | 2,000 | 2,000 | n/a | n/a |

The platform for integrating LonWorks compatible third-party devices consists of:

- System controller PXC00.D and automation station PXC50.D, PXC100.D or PXC200.D for integrating LonWorks devices via BACnet/LonTalk
- System controller PXC00-E.D and automation station PXC50-E.D, PXC100-E.D or PXC200-E.D for integrating LonWorks devices via BACnet/IP
- PXX-L11 and PXX-L12 expansion modules for 60 and 120 LonWorks devices

PX KNX

PX KNX connects KNX networks with Desigo and maps the group addresses to BACnet datapoints. PX KNX can handle the following main tasks:

- Data compression on the automation level (group functions)
- Time control
- Alarming, device monitoring
- Trend storage
- Mapping the Desigo RXB applications to BACnet for operating and monitoring

PX KNX supports the integration of:

- KNX S mode third-party devices
- RDF, RDG and RDU room thermostats
- RXB room automation stations

The PXC001.D system controller can integrate KNX via BACnet/LonTalk. The PXC001-E.D system controller can integrate KNX via BACnet/IP. PX KNX is preinstalled on PXC001..D controllers.

PX Modbus

PX Modbus connects Modbus devices or networks supporting the Modbus protocol to the Desigo system and maps their data points to BACnet data points. PX Modbus is particularly suitable for integrating industrial controls or chillers and linking them to the automation process.

The PXC001.D system controller can integrate Modbus via BACnet/LonTalk. The PXC001-E.D system controller can integrate Modbus via BACnet/IP. The PXA40-RS1 and PXA40-RS2 option modules support additional data points.

See *PX Modbus* (CA2N9772).

PX M-Bus

PX M-Bus connects the M-Bus consumption meters to the Desigo system and maps meter readings and device-related meter information to BACnet data points. PX M-Bus handles the following main activities:

- Measurement of consumption data and remote monitoring of max. 250 consumption and heat meters
- Compression of data from consumption and heat meters at the automation level
- Alarm handling, device monitoring
- Trend storage to record meter readings

The PXC001.D system controller can integrate M-Bus via BACnet/LonTalk. The PXC001-E.D system controller can integrate M-Bus via BACnet/IP. The PXA40-RS1 and PXA40-RS2 option modules support additional data points.

See *PX M-Bus* (CM2N9774).

PX SCL

PX SCL lets you quickly develop simple protocol solutions. The script control language from XWP is used with an interpretable environment and lets engineers create a solution. The solution cannot be used for complex protocols and solutions. It is used to develop other applications, such as local serial printer driver and pager applications.

The PXC001.D system controller can integrate SCL via BACnet/LonTalk. The PXC001-E.D system controller can integrate SCL via BACnet/IP. The PXA40-RS1 and PXA40-RS2 option modules support additional data points.

The regional companies develop the necessary protocols themselves.

The hotel management system Fidelio can be integrated into Desigo via PX SCL.

See *PX SCL* (CA2N9773).

PX LON

PX LON connects LonWorks networks to Desigo and maps Standard Network Variables (SNVT) to BACnet data points. The main functions of PX LON are:

- Higher-level control and optimization functions, such as room and zone-based groups, time control, and system functions, such as changeover, summer/winter compensation, etc.
- Alarm handling, device monitoring
- Trend storage

The PXC00.D system controller and the PXC50.D, PXC100.D and PXC200.D automation stations can integrate LonWorks devices via BACnet/LonTalk. The PXC00-E.D system controller and the PXC50-E.D, PXC100-E.D and PXC200-E.D automation stations can integrate LonWorks devices via BACnet/IP. With the PXX-L11 and PXX-L12 expansion modules you can connect 60 and 120 devices.

PX Open Platform SDK

HQ provides the PX Open Platform Software Development Kit (SDK) for experts in the regional companies.

21.3 Integration on field level

The integration of third-party devices and systems on the field level is appropriate:

- For communicative pumps, meters, etc.
- For small numbers of data points (10 to 100/160 data points)

TX Open is suitable for the integration of a few data points (from 10 to 160 data points). These data points can be processed further in the automation system or used for visualization in Desigo CC.

- The TXI1.OPEN module supports up to 100 data points and has an RS232/RS485 interface.
- The TXI2.OPEN module supports up to 160 data points and has in addition an ethernet connection for remote access, diagnosis and remote engineering.
- The TXI2-S.OPEN module supports up to 40 data points.

The TXI1.OPEN or TXI2.OPEN module is loaded with the protocol applications for Modbus/M-Bus/GENIbus/G120P and then works as the Modbus/M-Bus/US/GENIbus master. The values of the Modbus/M-Bus/GENIbus/G120P data points and the status of the existing data connection with the data points are transmitted to the automation station via the island bus and mapped to BACnet objects in the automation station. This way, the Modbus/M-Bus/GENIbus/G120P data points can be made available to all the devices and applications in the Desigo system.

The PXC50..D, PXC100..D and PXC200..D automation stations support TX Open. You can attach up to five TX Open modules to one PXC automation station.

Xworks Plus (XWP) is used to engineer all solutions. Various compounds are available, e.g., for pumps, variable speed drives and heat meters.

Predefined solutions allow for simple commissioning. Solutions for Grundfos, Wilo, Danfoss and G120P are delivered as part of the HQ CAS Library. For M-bus and Modbus, sample solutions serving as device description templates are provided (TX Open templates).

The following solutions on the TX Open platform are available:

- TX Modbus
- TX M-Bus
- TX G120P/SED2
- TX Grundfos via GENIbus
- TX Open platform (SDK)

TX Modbus

TX Modbus supports Modbus RTU and Modbus TCP, Wilo pumps and variable speed drives. TXI2.OPEN supports 160 data points. They may be distributed in any fashion to the devices for the Modbus system. The number of devices is only limited by the 160 data points.

See *TX Modbus Engineering Guide* (CM110571).

TX M-Bus

TX M-Bus supports templates for meters. The regional companies can create templates. You need a level converter for TX M-Bus. TXI2.OPEN supports 160 data points. They may be distributed in any fashion to the devices for the M-bus system. The number of devices is only limited by the 160 data points.

See *TX M-Bus Engineering Guide* (CM110572).

TX G120P

TX G120P supports the integration via the Modbus and USS protocol. You can integrate up to eight G120P variable speed drives per TX Open module into the Desigo system.

See *TX G120P Engineering Guide* (CM110576).

TX SED2

TX SED2 supports the integration via the USS protocol. You can integrate up to eight SED2 variable speed drives per TX Open module into the Desigo system.

You can add new G120P variable speed drives to an existing TX Open (USS) with already installed SED2 drives, e.g., when:

- A defective SED2 needs to be replaced with a G120P in an existing project
- An existing project with installed SED2 drives needs to be extended with a new G120P

See *TX SED2 Engineering Guide* (CM110573).

TX Grundfos via GENibus

TX Grundfos supports the integration of Grundfos via GENibus. You can integrate up to eight Grundfos pumps into the Desigo system.

See *TX Grundfos / GENibus Engineering Guide* (CM110574).

TX Open Platform (SDK)

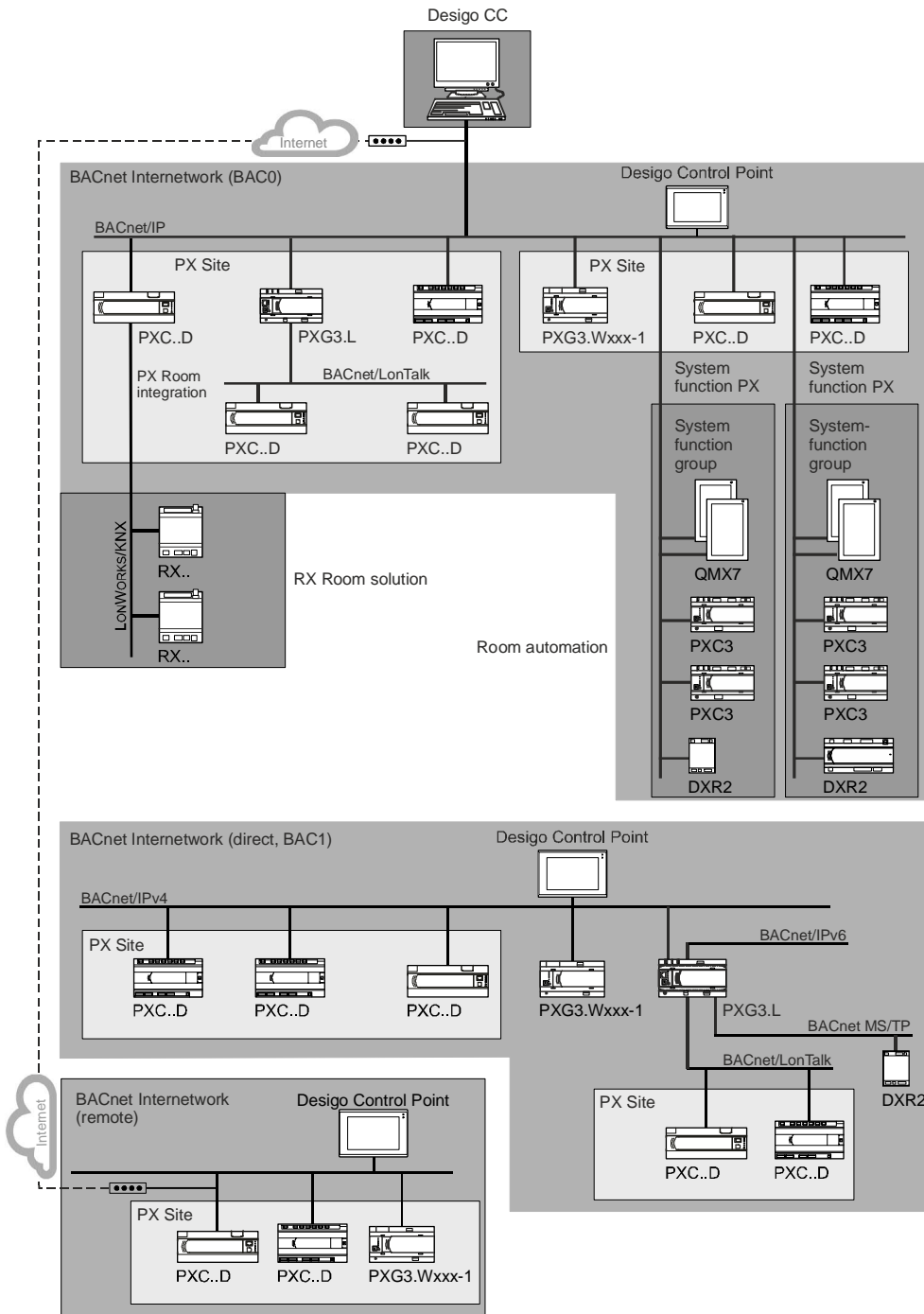
HQ provides the TX Open Platform Software Development Kit (SDK), including training, for experts in the regional companies. The training course provides the necessary tools and knowledge to create new protocol applications.

21.4 Integration on room level

See Network architecture [→ 179].

22 System configuration

System overview



Designo system

Covers all the devices on the MLN (Management Level Network), ALN (Automation Level Network) and FLN (Field Level Network).

One Designo system may comprise several BACnet internetworks. These are connected into a system with Designo CC. In this case, Designo CC appears as a BACnet device in several BACnet internetworks.

BACnet internetwork

A BACnet internetwork consists of one or several BACnet networks. Individual BACnet networks are connected to BACnet routers.

Each BACnet device can communicate with another BACnet device in the internetwork. A BACnet device in one internetwork cannot communicate with a device in another internetwork.

A Desigo management station can be used to integrate the operation of several BACnet internetworks and other systems (see Desigo system).

When defining the system configuration, FLN integrations (LonWorks, KNX) are also added to the BACnet internetwork. In this way, the Desigo system can be seen as a combination of several BACnet internetworks. Technically, the individual FLN devices are not BACnet devices. They do not communicate via the BACnet protocol.

BACnet PTP internetwork

BACnet PTP communication uses modem (telephony) or null-modem (RS232) connections. Owing to the slow rate of data transfer via these connections, the limits are lower for a BACnet PTP internetwork. Modem-based PTP connections are considered obsolete and are therefore no longer used. The BACnet PTP communication connects BACnet networks via BACnet half routers.

BACnet network

A quantity of BACnet devices connected within an IP or LonTalk or MS/TP network with specific (that means, the devices are in the same BACnet Broadcast Domain) limits. In the case of the LonTalk or MS/TP network, the limit is physical. In the case of an IP network, the network can be physically the same, but the limit is determined by different UDP ports.

Local communication between two BACnet devices in a BACnet network is not visible in another BACnet network.

IP segment

Sub-area of an IP network. IP segments are connected by IP routers.

In order to ensure that BACnet communications (Broadcasts) can always take place across IP routers, BBMDs (BACnet Broadcast Management Devices) are required. PXG3.L/M and PXC...-E.D over IP can be configured as BBMDs. Individual BACnet devices in an IP segment can register with a BBMD as foreign devices.

LonWorks segment (ALN)

Sub-area of a BACnet/LonTalk network. LonWorks segments are connected by LonWorks routers. In most cases it is not necessary to divide a BACnet/LonTalk network into several LonWorks segments (ALN).

It is not possible to use a LonWorks router because of the restricted length of the data packets. An L-Switch can be used as a router on the ALN.

LonWorks segment (FLN)

Sub-area of a LonWorks network. LonWorks segments are connected by LonWorks routers.

An L-Switch or a LonWorks router can be used as a router on the FLN.

LonWorks trunk (FLN)

Comprises all the devices connected on the FLN side of the PXC00.D/-E.D + PXX-L1.... Consists of one or several LonWorks segments (FLN).

A LonWorks trunk (FLN) is the equivalent of a LonWorks network (FLN).

PX KNX integration

Comprises the integration of KNX devices that are connected on the FLN side of the PXC001.D/-E.D.

PX site

A Desigo PX automation system site.

The PX BACnet devices which control the plant in a PX site are interconnected via the global objects and the primary copy procedure.

A PX site is independent of the limits affecting the BACnet network. A site can extend over several BACnet networks. One BACnet network may include several sites. All the associated limits must be maintained simultaneously.

A PX site cannot be extended beyond the limits of a BACnet internetwork. This is particularly important in the case of BACnet PTP internetworks.

PX plant

A PX plant is part of a PX site and generally comprises several partial plants (plant structure).

A PX plant can be distributed over several PX BACnet devices. In principle PX BACnet devices can be distributed to different BACnet networks. However, owing to the communications load between partial plants, this is not recommended.

The plant structure is mapped to BACnet by means of hierarchy objects. Operator units with generic operation automatically read this structure.

BACnet MS/TP

A BACnet MS/TP network is a BACnet network that is physically based on EIA-485 and operated using a BACnet-specific MasterSlave/TokenPassing data link protocol (see BACnet standard clause 9). An MS/TP network is linked via a BACnet router to a BACnet/IP or BACnet/LonTalk network.

Desigo room automation

Includes the BACnet devices connected directly to BACnet/IP or BACnet MS/TP, used for room automation.

These BACnet devices are not part of a PX site. There is no connection via global objects and the primary copy procedure.

Desigo room automation system functions

In Desigo room automation, primary subsystem control functions are centralized as Desigo room automation system functions.

PX system functions

A PXC.. of a PX site as PX system function can assume Desigo room automation subsystem functions such as scheduling, life check, time synchronization for a Desigo room automation system function group for BACnet devices for room automation.

System function group

A Desigo room automation system function group cannot be identified or defined via the network topology. Engineering the Desigo room automation system functions of the PX system functions determines the Desigo room automation system function group.

For more information, see chapter *System Overview* and *Network Architecture*.

22.1 Technical limits and limit values

There are two types of limits:

- Technical limits (hard-coded limits) are maintained by technical means. They cannot be exceeded.
- Recommended limits (soft limits) are not enforced by technical means. They can be exceeded.

The limits are defined to ensure the full and correct functioning of the system. Consult Headquarters before exceeding the recommended limits. HQ can modify the recommended limits on the basis of new findings at any time. Changes are notified in Facts bulletins.

Certain limits cannot be verified (for reasons of cost or quantity). These limits are shown in this document as follows:

| This limit type... | ...is shown like this | Example |
|---|-----------------------|-------------------|
| Technical limit verified | Limit* | 60* |
| Technical limit NOT verified | [Limit*] | [50*] |
| Recommended limit verified | Limit | 64 |
| Recommended limit NOT verified | [Limit] | [1'000] |
| Limit subject to proviso (refer to footnotes) | (Limit) | (10) ⁹ |

22.2 Maximum number of elements in a network area

| Number of elements / Per network area | Desigo system | BACnet inter-network | BACnet PTP inter-network | BACnet/IP network | BACnet MS/TP network | BACnet/LonTalk network | LonWorks trunk (FLN) | PX KNX integration | PX site |
|--|------------------------|-----------------------|--------------------------|--|----------------------|------------------------|----------------------|--------------------|------------------------|
| Desigo topology | | | | | | | | | |
| BACnet internetwork | 200 | | n/a | n/a | n/a | n/a | n/a | n/a | n/a |
| BACnet PTP internetwork | 1 ¹⁸ | n/a | | n/a | n/a | n/a | n/a | n/a | n/a |
| BACnet/IP network | n/a | 10 [50*] | [1] ¹⁹ | | n/a | n/a | n/a | n/a | [total 20] |
| BACnet MS/TP network | n/a | [50] | n/a | 8 | | n/a | n/a | n/a | n/a |
| BACnet/LonTalk network | [3] | n/a | n/a | n/a | n/a | | n/a | n/a | |
| PXG3.M/L (BACnet router) | n/a | [100] | [30] | [100] | 1 | 1 | n/a | n/a | n/a |
| IP segment | n/a | 10 [50*] ⁶ | [50*] ⁶ | 10 [50*] ⁶ | n/a | n/a | n/a | n/a | n/a |
| LonWorks segment (ALN) | n/a | [100] | [30] | n/a | n/a | 1 | n/a | n/a | n/a |
| PX site | [1,000] | [30] | 5 | n/a | n/a | n/a | n/a | n/a | n/a |
| PX plant | [4,000] | [2,000] | [60] | n/a | n/a | n/a | n/a | n/a | 100 |
| LonWorks trunk (FLN) | [200] | [100] | [30] | n/a | n/a | n/a | | n/a | [50] |
| LonWorks segment (FLN) | n/a | n/a | n/a | n/a | n/a | n/a | [5] | n/a | [250] |
| PX KNX integration | [200] | [100] | [30] | n/a | n/a | n/a | n/a | | [50] |
| Desigo devices | | | | | | | | | |
| PX... without DXR2/PXC3 ¹⁵ | [2,000] | [1,000] ⁹ | [30] | [200] ⁸ | n/a | 30 | n/a | n/a | 50/100 ¹⁷ |
| PXC3.E (Desigo room automation) | (10,000) ¹⁶ | [1,500] (2,500) | n/a | 250 ²³ [500] ²⁴ | n/a | n/a | n/a | n/a | n/a ¹⁵ |
| DXR2 ¹⁴ (Desigo room automation) | (10,000) ¹⁶ | [1,500] (2,500) | n/a | 250 ²³ [500] ²⁴ | 64 ²¹ | n/a | n/a | n/a | n/a ¹⁵ |
| DXR1 (Desigo room automation) | (10,000) ¹⁶ | [1,500] (2,500) | n/a | 100 ²³ [500] ²⁴ | 64 ²¹ | n/a | n/a | n/a | n/a ¹⁵ |
| Desigo CC | 10 | 10 | n/a | 10 | n/a | n/a | n/a | n/a | n/a |
| PXM20 | n/a | (10) ⁹ | 10 | n/a | n/a | 10 | n/a | n/a | total 15 ¹⁰ |
| PXM30/40/50.E | n/a | n/a | n/a | 10 | n/a | n/a | n/a | n/a | n/a |
| PXG3.W100-1 | n/a | n/a | n/a | 10 | n/a | n/a | n/a | n/a | n/a |
| PXG3.W200-1 | n/a | n/a | n/a | 10 | n/a | n/a | n/a | n/a | n/a |
| Desigo Xworks Plus (XWP) (commissioning) ¹² | n/a | [10] | [10] | [10] | n/a | [5] | n/a | n/a | total 15 ¹⁰ |
| Total LonWorks nodes | [40,000] | [20,000] | [6,000] | n/a | n/a | n/a | 300 ² | n/a | [10,000] |
| RXB | [8,000] | [2,000] | [1,200] | n/a | n/a | n/a | n/a | 45 ⁴ | [2,000] |
| Data points and BACnet objects | | | | | | | | | |
| Physical data points | [100,000] | [100,000] | [3,000] | [20,000] | n/a | n/a | n/a | n/a | [6,000] |
| Total BACnet objects | [500,000] | [100,000] | [30,000] | [100,000] | [3,000] | [30,000] | n/a | n/a | [50,000] |
| Trendlog object | [30,000] | [2,500] | [200] | [2,500] | [540] | [600] | n/a | n/a | [1,000] |

Key

n/a Not applicable.

- No restrictions.

² Not the same as the number of integrated devices (PXC00.D/-E.D + PXX-L...).

³ LonWorks routers must not be used at the automation level.

- ⁴ Limit applies only if this device type is used exclusively.
- ⁶ For higher performance use a PXG3 instead of a PXC-E.D for the BBMD function.
- ^{7a} Limit for PXC00.D/-E.D + PXX-L11.
- ^{7b} Limit for PXC00.D/-E.D + PXX-L12.
- ⁸ Limit for PX devices without Desigo room automation. Do not exceed the number of PXC devices per site.
- ⁹ The limit on the number of PX automation stations per internetwork can only be maintained if no PX clients are used. PX clients limit the permissible number of PX per internetwork. The values can be obtained by reference to the relevant automation station columns. The restricted view option does not affect the system configuration of PX clients.
- ¹⁰ The number of temporary alarm receivers in a PX is a technical limit. The recommended limit is lower. This takes account of the fact that additional devices may be connected for service purposes.
- ¹¹ The number of temporary alarm receivers in a PX is a technical limit. The recommended limit is lower. This takes account of the fact that additional alarm receivers (third-party) may have entries in this list.
- ¹² Parallel engineering (commissioning) is possible subject to the following restrictions:
- Node setup: Only one XWP per LonTalk/IP segment.
 - Download and online operation: only one XWP for each automation station.
- ¹⁴ In pressurized rooms with or without fume hood, all automation stations of a room must be connected to a switch in a starlike manner to ensure high availability.
- ¹⁵ Desigo room automation stations do not belong to a PX site (no primary copy function).
- ¹⁶ For the system configurations of the Desigo CC management platform, see *Desigo CC System Description (A6V10415500)*.
- ¹⁷ 50: If Lon PX exists in the PX site. 100: If no Lon PX exist in the PX site (only IP PX).
- ¹⁸ These limits in the Desigo system refer in particular to Desigo Insight. The limits may be significantly lower due to the PTP connection(s) outside the Desigo system and their technical limitations. Examples of such limitations outside the Desigo system can include available bandwidth for the PTP link or available modem speeds.
- ¹⁹ This limit can be exceeded if all BACnet devices are located within the same IP subnetwork, or if no communication between the various BACnet/IP networks is required.
- ²⁰ These limits apply only to IP-based DXR2 devices.
- ²¹ These limits apply to MS/TP-based DXR2 devices.
- ²³ Multiple IP segments per BACnet internetwork.
- ²⁴ One IP segment per BACnet internetwork.

For more information about networks, see *Application Guide for IP Networks in Building Automation Systems (CM110668)*.

22.3 Desigo room automation system function group limits

A Desigo room automation system function group comprises parts of the Desigo room automation stations on the BACnet internetwork. Grouping occurs based on Desigo room automation stations assignment to PX system function responsible for the Desigo room automation subsystem functions.

Desigo room automation defines Desigo room automation system functions comprising life check and time synchronization.

The current limits for the Desigo room automation system function group are mainly imposed by life check and scheduling carried out by the Desigo room automation system function PX.

A PXC3 generally controls several (about 5..8) multiple rooms. The number of rooms in the Desigo room automation system function group are the decisive factor for some limits.

Desigo room automation stations are not part of a PX site. Data are not aligned between the primary PX site and the Desigo room automation stations.

Desigo room automation stations do not support BBMDs. This restricts the BACnet/IP network, that is, all Desigo room automation stations and their Desigo room automation system function PX or a PXG router must be located in the same IP segment.

Desigo room automation stations with own alarming

| Item | Limit | Description |
|---|-------|--|
| Trend per room | 5 | It is assumed that a maximum of 5 trend points are logged on average. Assumption for the trend interval: 15 minutes. |
| Number of external BACnet references | 500 | Maximum number of external BACnet references that support a Desigo room automation system function PX. The Desigo room automation system function PX requires external references for the life check and for scheduler functions. Examples of objects with external references: - EventEnrollment: 1reference - Schedule: 1-5 references |
| Event Enrollment per Desigo room automation station | 1 | Number of Event Enrollment objects required on the Desigo room automation system functions PX for the life check per Desigo room automation station. |
| Sample number of Desigo room automation stations per Desigo room automation system function group | 250 | Desigo room automation system function PX with maximum scheduler functions. The limit designates the maximum number of Desigo room automation stations in the Desigo room automation system function group. In this example, it is assumed that the following scheduler objects are available on the Desigo room automation system function PX: - Maximum number of scheduler objects - Per scheduler object, maximum number of external references |
| Sample number of Desigo room automation stations per Desigo room automation system function group | 500 | Desigo room automation system function PX without scheduler functions. The limit designates the maximum number of Desigo room automation stations on the Desigo room automation system function group. In this example, it is assumed that no scheduler objects are available on the Desigo room automation system function PX. |

22.4 Devices

22.4.1 PXC..D automation stations / system controllers

| Item | PX Compact | | PX Modular | | | | PX Open ¹⁰ | PX KNX ⁹ |
|--|--|--|----------------------|------------------------|------------------------|----------------------|--|------------------------|
| | PXC12.D PXC12-E.D PXC22.D PXC22-E.D PXC-NRUF | PXC22.1.D PXC22.1-E.D PXC36.1.D PXC36.1-E.D | PXC50.D PXC50-E.D | PXC100.D PXC100-E.D | PXC200.D PXC200-E.D | PXC00.D PXC00-E.D | PXC001.D PXC001-E.D + PXA40- RS.. | PXC001.D PXC001-E.D |
| Temporary alarm receiver ¹ | 18* | 18* | 18* | 18* | 18* | 18* | 18* | 18* |
| Configured alarm receivers ² | 20* | 20* | 20* | 20* | 20* | 20* | 20* | 20* |
| BACnet references COV server resources ³ | [1,400*] | [1,400*] | [1,400*] | [1,400*] | [1,400*] | [1,400*] | [1,400*] | [1,400*] |
| BACnet references COV client resources ⁴ | 400* | 950* | 950* | 950* | 950* | 950* | 650* | 400* |
| Total BACnet objects | [4,000] | [4,000] | [4,000] | [4,000] | [4,000] | [4,000] | [4,000] | [4,000] |
| Number of function block | 1,900* | 1,900* | 1,900* | 1,900* | 2,900* | 1,900* | 2,900* | 2,900* |

| Item | PX Compact | | PX Modular | | | | PX Open ¹⁰ | PX KNX ⁹ |
|---|--|--|----------------------|------------------------|------------------------|----------------------|--|------------------------|
| | PXC12.D PXC12-E.D PXC22.D PXC22-E.D PXC-NRUF | PXC22.1.D PXC22.1-E.D PXC36.1.D PXC36.1-E.D | PXC50.D PXC50-E.D | PXC100.D PXC100-E.D | PXC200.D PXC200-E.D | PXC00.D PXC00-E.D | PXC001.D PXC001-E.D + PXA40- RS.. | PXC001.D PXC001-E.D |
| instances (application size) | | | | | | | | |
| Trend log ⁵ | 100 | 100 | 100 | 200 | 350 | 200 | 600 | 100 |
| Trend log multiple ¹⁹ | 20 | 20 | 20 | 20 | 20 | 20 | 20 | 20 |
| Scheduler | 15 ¹⁷ | 15 ¹⁷ | 50 ¹⁸ | 50 ¹⁸ | 50 ¹⁸ | 50 ¹⁸ | 50 ¹⁷ | 50 ¹⁷ |
| Calendar ¹⁴ | 10 | 10 | 50 | 50 | 50 | 50 | 50 | 50 |
| PXM10 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | n/a |
| PPS2 devices (ALN) ⁸ (e.g. QAX3.x, RXZ90.1) | 5 | 5 | n/a | n/a | n/a | n/a | 5 | n/a |
| Physical data points onboard | 12 / 22 / 64 | 22 / 36 | n/a | n/a | n/a | n/a | n/a | n/a |
| Physical data points TX-I/O | n/a | 16 / 28 | 80* | 200* ¹⁵ | (350) ¹⁵ | n/a | n/a | n/a |
| Physical data points onboard + TX-I/O | n/a | 38 / 64 | 80 | n/a | n/a | n/a | n/a | n/a |
| Physical data points TX-I/O + PTM | n/a | n/a | 52* | 200* ¹⁵ | (350) ¹⁵ | n/a | n/a | n/a |
| Total number of data points onboard + TX- I/O + PTM + TX Open | n/a | 400* | 400* | 600* ¹⁵ | (1,000) ¹⁵ | n/a | n/a | n/a |
| TX Open per island bus | n/a | 5 | 5 | 5 | 5 | n/a | n/a | n/a |
| PXX-PBUS | n/a | n/a | 1 | 1 | 1 | n/a | n/a | n/a |
| Dynamic calendar objects ²⁰ | 10* | 10* | 10* | 10* | 10* | 10* | 10* | 10* |
| Dynamic event enrollment objects ²⁰ | 50* | 50* | 50* | 50* | 50* | 50* | 50* | 50* |
| Dynamic notification class objects ²⁰ | 50* | 50* | 50* | 50* | 50* | 50* | 50* | 50* |
| Dynamic schedulers | 10* | 10* | 10* | 10* | 10* | 10* | 10* | 10* |
| Dynamic trend log objects ²⁰ | 100* | 100* | 100* | 100* | 100* | 100* | 100* | 100* |
| Dynamic trend log multiple objects ²⁰ | 20* | 20* | 20* | 20* | 20* | 20* | 20* | 20* |

Key

- ¹ PXM20, PX-Web and XWP are temporary alarm receivers.
- ² Desigo CC is a configured alarm receiver. The number of entries in the notification class is limited to 20. The total number of different configured alarm receivers across all notification classes is limited to 30.
- ³ Max. number of BACnet references, COV servers: SubscribeCOV requests which can be accepted. Example: 1400: 1 client and 1400 values or 2 clients and 700 values.
- ⁴ Max. number of BACnet references, COV client resources, i.e. values read from or written to (commanded) your own automation station or a remote automation station.
BACnet client references are used in Input, Output, Scheduler, Trendlog and Group objects (all NameRef_Type inputs with AddrKind = B). The configured alarm receivers of the Notification Class objects do NOT require any BACnet client references.
The available number of BACnet client references shall address not more than 50 different remote automation stations. If this value is exceeded the number of BACnet broadcast messages on the network will increase.
Example: 400: 1 client and 400 values or 2 clients and 200 values.
- ⁵ Every active Trendlog object needs a BACnet reference.
Trends need 12 bytes per entry (irrespective of data type). Max. 64 KB can be allocated to the log buffer (approx. 5,000 entries) for each Trendlog object. These log buffers are assigned in D-MAP RAM. If the log buffer size is changed and there is insufficient D-MAP RAM available, the Reliability property of the Trendlog object is set to Memory limit reached.
- ⁸ The address of the PPS-2 devices QAX84.1 and RXZ90.1 is always 1 (no address selection).
- ⁹ PX KNX = PXC001.D / PXC001-E.D and loaded PX KNX firmware.
- ¹⁰ PX Open = PXC001.D / PXC001-E.D with option module PXA40-RS1/RS2 and loaded PX Open firmware.
- ¹⁴ Maximum 30 calendar entries.
- ¹⁵ The number of physical data points influences the reaction time of the application. If minimum reaction times are specified, the number of physical data points may have to be reduced.
The following relationship between reaction times and the number of physical data points can be assumed:
- up to 150 physical data points = Reaction times < 1s
- up to 250 physical data points = Reaction times 1...2 s
- up to 350 physical data points = Reaction times 2...3 s
- ¹⁷ Number of switching times per day: 10; max. 5 BACnet references.
- ¹⁸ Number of switching times per day: 20; max. 5 BACnet references.
- ¹⁹ Every active trendlog multiple object needs a BACnet reference per logged value.
5 logged values are assumed for the number of trendlog multiple objects (number of Trendlog / 5).
Trends need 12 bytes per entry (irrespective of data type).
Max. 64 KB can be allocated to the log buffer (approx. 5,000 entries) for each trendlog object.
These log buffers are assigned in D-MAP RAM.
If the log buffer size is changed and there is insufficient D-MAP RAM available, the Reliability property of the Trendlog object is set to Memory limit reached.
- ²⁰ Dynamic objects are counted the same as non-dynamic objects for total limits.
- * Limit* = Technical limit verified.
- [*] [Limit*] = Technical limit not verified.
- [] [Limit] = Recommended limit not verified.
- () (Limit) = Limit subject to proviso. Refer to footnotes.

D-MAP RAM

If the whole D-MAP RAM is taken up with trendlog objects, a delta (differential) download will no longer be possible.

The overall size of the free and used D-MAP RAM can be viewed with XWP, Desigo CC or PXM20. The information concerned is stored in the device object under the memory statistics property [MemStc].

Access rights management

Access rights are managed via USPRF. You can define a maximum of 10 user groups and 20 users. 10 user groups and 6 users are already predefined as a template (global chart).

22.4.2 LonWorks system controllers

Device combination: PXC00.D/-E.D + PXX-L11/12

| Item | Limit |
|---|---|
| LonWorks devices: PXX-L11 PXX-L12 | 60* (e.g.: 5 Group Members are defined, that means 5 x 12 = 60 COV resources are needed) 120* Max. number of integrated LonWorks devices covers third-party LonWorks devices. |
| Discipline I/Os | [600] max. number of discipline I/O objects |
| Groups | [50] max. Number of groups |
| Room members | No limits |
| Group members | Cross-disciplinary groups can have more than 5 destinations. The number of cross-disciplinary groups depends on the COV client resources (max. 250). A different number of COVs is required, depending on the group type. These must be multiplied by the number of destinations. |

Calculation basis:

| | | |
|----------------|----------|---|
| HVAC | CHGRP | 1 COV client resource per destination |
| | SPGRP* | 12 COV client resources per destination |
| | EMGRP | 1 COV client resource per destination |
| Lighting | LIGHTGRP | 2 COV client resources per destination |
| Blinds | BLSGRP | 4 COV client resources per destination |
| Building use | USEGRP | 3 COV client resources per destination |
| Room occupancy | OCGRP | 3 COV client resources per destination |

LonWorks system controllers with physical I/Os and TX Open

Device combination: PXC50/100...D + PXX-L11/12

If the PXC50/100...D is used instead of the PXC00...D as a system controller, physical I/Os can be integrated via TX-I/O modules and TX Open data points. The reaction time can be greater for a larger number of physical I/Os or TX Open data points, and depending on the complexity of the CFC program.

22.4.3 Automation stations with LonWorks integration

Device combination: PXC50/100/200...D mit PXX-L11

The modular automation stations PXC50/100/200.D and PXC50/100/200-E.D allow the integration of LonWorks devices and third-party devices via PXX-L11 in addition to the use of I/O modules or third-party devices via TX Open.

The integration on PXC50...D is limited to a maximum of 10 LonWorks devices.

The integration of LonWorks devices for PXC100/200...D is limited by response times.

The following values can be assumed for reaction times depending on the number of physical data points:

| Reaction times depending on number of physical data points | Without LonWorks devices | Up to 5 LonWorks devices | 5 to 20 LonWorks devices |
|--|--------------------------|--------------------------|--------------------------|
| Max. 150 data points | < 1s | 1-2s | 3-4s |
| Max. 250 data points | 1-2s | 2-3s | 4-5s |
| Max. 350 data points | 2-3s | 3-4s | 5-6s |

22.4.4 PX Open integration (PXC001.D/-E.D)

| Item | Limit | Description |
|--------------------|--------|---|
| Modbus data points | [250*] | Max. number of data points per PX Modbus. |
| SCL data points | [250*] | Max. number of data points per PX SCL. |
| M-bus data points | [250*] | Max. number of data points per PX M-bus. |
| M-bus meters | [250] | Max. number of M-bus meters in PX M-bus applications. |

22.4.5 PX Open integration (PXC001.D/-E.D + PXA40-RS1)

| Item | Limit | Description |
|--------------------|--------|---|
| Modbus data points | [800*] | Max. number of data points per PX Modbus. |
| SCL data points | [800*] | Max. number of data points per PX SCL. |
| M-bus data points | [800*] | Max. number of data points per PX M-bus. |
| M-bus meters | [250] | Max. number of M-bus meters in PX M-bus applications. |

22.4.6 PX Open integration (PXC001.D/-E.D + PXA40-RS2)

| Item | Limit | Description |
|--------------------|----------|---|
| Modbus data points | [2,000*] | Max. number of data points per PX Modbus. |
| SCL data points | [1,000*] | Max. number of data points per PX SCL. |
| M-bus data points | [2,000*] | Max. number of data points per PX M-bus. |
| M-bus meters | [250] | Max. number of M-bus meters in PX M-bus applications. |

22.4.7 PX KNX integration (PXC001.D/-E.D)

These limits also apply to PXC00-U.

The maximum number of devices only applies in cases where only one device type is used. The following formula applies to mixed operation with third-party devices: $50 * \text{RXB} + \text{third-party devices} < 2,000$ data points.

| Item | Limit | Description |
|---------------------|----------|--|
| KNX/EIB data points | [2,000*] | Max. number of KNX data points that can be integrated (KNX communication objects). |
| RXB | 45 | Max. number of RXB devices per KNX (approx. 50 KNX data points per RXB, depending on the application). |

22.4.8 TX Open integration (TXI1/2/2-S.OPEN)

| Item | Limit | Description |
|-------------|-------|---|
| TXI1.OPEN | 100* | Max. number of data points per TX Open. |
| TXI2.OPEN | 160* | Max. number of data points per TX Open. |
| TXI2-S.OPEN | 40* | Max. number of data points per TX Open. |

22.4.9 Number of data points on Desigo room automation stations

Number of data points on the TX-I/O subsystem

Every used data point on TX-I/O is counted.

| ASN | Product description | Data points | Description |
|----------|--------------------------------------|-------------|--|
| TXM1.6RL | 6 I/O relay modules, bistable | max. 6 | Used TX-I/Os are counted. |
| TXM1.8RB | 8 I/O blinds modules | max. 8 | Used TX-I/Os are counted (1 data point per relay). |
| TXM1.8T | 8 I/O triac modules | max. 8 | Used TX-I/Os are counted. |
| TXM1.8U | 8 I/O universal modules (DI, AI, AO) | max. 8 | Used TX-I/Os are counted. |
| TXM1.6R | 6 I/O relay modules | max. 6 | Used TX-I/Os are counted. |
| TXM1.8D | 8 I/O digital input modules | max. 8 | Used TX-I/Os are counted. |
| TXM1.16D | 16 I/O digital input modules | max. 16 | Used TX-I/Os are counted. |

Number of data points on DALI subsystem

Each individually controlled DALI lighting group and each individually controlled ECB counts as 1 data point.

| ASN | Product description | Data points | Description |
|-----------------------------|---------------------|-------------|---|
| PXC3.E7xA PXC3.E16A-200A | Automation station | max. 64 | DALI lighting groups and/or individual DALI ECBs are counted. |

Additional DALI limits:

- Max. number of devices: 64
- Max. number of addresses: 64
- Max. number of groups: 16

Number of data points on KNX PL-Link subsystem

KNX PL-Link devices have a set count whereas KNX S-Mode devices are counted according to the used group addresses.

| ASN | Product description | Data points | Description |
|----------|--|-------------|-------------|
| RXM21.1 | Fan coil PL-I/O | 5 | Fixed count |
| RXM39.1 | Fan coil PL-I/O | 5 | Fixed count |
| QMX2.P33 | Room operator unit with display and temperature sensor | 3 | Fixed count |
| QMX2.P43 | Room operator unit with display, temperature and humidity sensor | 4 | Fixed count |
| QMX3.P02 | Freely configurable operator unit, wall mounted | 5 | Fixed count |
| QMX3.P30 | Freely configurable operator unit, wall mounted | 1 | Fixed count |
| QMX3.P36 | Freely configurable flush-mounted room unit | 3 | Fixed count |

| ASN | Product description | Data points | Description |
|------------------------|--|-------------|---|
| QMX3.P34 | Freely configurable operator unit, wall mounted | 3 | Fixed count |
| QMX3.P37 | Freely configurable operator unit, wall mounted | 7 | Fixed count |
| QMX3.P40 | Room operator unit without display, with temperature and humidity sensor | 2 | Fixed count |
| QMX3.P70 | Freely configurable operator unit, wall mounted | 3 | Fixed count |
| QMX3.P74 | Freely configurable operator unit, wall mounted | 5 | Fixed count |
| QMX3.P87 | Freely configurable operator unit, wall mounted (fume hood) | 3 | Fixed count |
| QMX3.P88 | Freely configurable operator unit, flush-mounted (fume hood) | 3 | Fixed count |
| AQR253... AQR257... | Flush-mounted room sensor with: Front module Base module | 1-3 | Fixed count, 1 data point per measured value (optional potential-free, passive NTC sensors are not counted) |
| UP220/31 | Switch interface | 4 | Fixed count |
| UP221/x | Single switch | 2 | Fixed count |
| UP222/x | Double switch | 4 | Fixed count |
| UP223/x | Triple switch | 6 | Fixed count |
| UP287/x | Quadruple switch | 8 | Fixed count |
| UP258D1x | Occupancy, light sensor | 2 | Fixed count |
| UP255/D12 | Brightness sensor | 1 | Fixed count |
| RL260xx | 4 x binary input | 4 | Fixed count |
| RL512xx | 1 x light 16A | 1 | Fixed count |
| RL513xx | 3 x light 6A | 3 | Fixed count |
| RL521xx | 2 x blinds | 4 | Fixed count |
| RL526D23 | Switching/Dimming actuator 2 x AC 230 V, 10 A, 1 ... 10 V | 2 | Fixed count |
| RS510xx | 2 x light 10A | 2 | Fixed count |
| RS520xx | 1 x blind | 2 | Fixed count |
| RS525xx | 1 x light universal dim | 1 | Fixed count |
| UP285/x | 1 x switch | 2 | Fixed count |
| UP286/x | 2 x switch | 4 | Fixed count |
| UP287/x | 4 x switch | 8 | Fixed count |
| UP510/xx | 2 x light 10A | 2 | Fixed count |
| UP520/xx | 1 x blind | 2 | Fixed count |
| UP525/xx | 1 x light universal dim | 1 | Fixed count |
| N528D01 | Universal dimmer, 2 x 300 VA, AC 230 V | 2 | Fixed count |
| KNX S-Mode | Third-party device | | Group addresses used are counted |

Additional KNX PL-Link limitations:

- Max. number of devices:
 - 64 on PXC3.xx
 - 32 on DXR2.xx
- The range of the Individual Address (IA) can be defined as follows in Desigo room automation:

- KNX S-Mode: 1 ... 179
- KNXnetIP: 180 und 181
- KNX PL-Link devices: 182 ... 250
- Desigo room automation station: 251
- Max. number of KNX S-Mode group addresses: 238

22.4.10 Number of data points for PXC3

A PXC3.E72x supports max. 4 rooms or 8 room modules and is limited to 72 TX-I/O data points.

A PXC3.E.75 supports max. 8 rooms or 16 room modules and is limited to 200 TX-I/O data points.

These criteria must be satisfied to be able to select the correct PXC...:

- The physical TX-I/O data points used
- The total number of I/O data points used from TX-I/O, KNX PL-Link, and DALI

| ASN | Physical TX-I/O data points used | Total I/O data points (TX-I/O, DALI, KNX PL-Link) |
|-----------|----------------------------------|---|
| PXC3.E16A | n/a | 64 |
| PXC3.E72 | 72 | 140 |
| PXC3.E72A | 72 | 140 |
| PXC3.E75 | 200 | 280 |
| PXC3.E75A | 200 | 280 |

Web clients for room operation

| Item | Limit | Description |
|--|--------|--|
| Standard web clients ¹ | 8 | Recommended number of web clients that can simultaneously access a PXC3. |
| Templates with standard background pictures ² | 6 | Maximum number of different templates which are using the default background pictures. |
| Customized background pictures ² | 1.5 MB | Maximum total size of all customized background pictures (the PNG file format is used as a reference). |

Key

¹ Restriction: When using standard web clients (web browser on PCs, smart phones, tablets, etc.), the screen display and operation (touch or mouse) are neither modified nor tested for the available browsers.

² Valid values when using 8 room applications at the boundary of maximum system limits.

22.4.11 Number of data points for DXR1

| ASN | Max. number of data points | Description |
|-----------------|----------------------------|------------------------|
| DXR1.E04PDZ-112 | n/a | 2 UI |
| DXR1.E09PDZ-112 | 16 | 2 UI, 4 DO, 1 AO |
| DXR1.E09PDZ-113 | 16 | 2 UI, 4 DO, 1 AO |
| DXR1.E10PL-112 | 22 ^{a2} | 2 UI, 1 DI, 4 DO, 1 AO |
| DXR1.E10PL-113 | 19 ^{b2} | 2 UI, 1 DI, 4 DO, 1 AO |

| ASN | Max. number of data points | Description |
|-----------------|----------------------------|------------------|
| DXR1.E02PLZ-112 | 2 | n/a |
| DXR1.M04PDZ-112 | n/a | 2 UI |
| DXR1.M09PDZ-112 | 16 | 2 UI, 4 DO, 1 AO |
| DXR1.M09PDZ-113 | 16 | 2 UI, 4 DO, 1 AO |

Key

- ^{a1} Total Number of PL-Link data points (under unconfigured status): 36
- ^{a2} Activate all selectable PL-Link devices, then check the maximum number of PL-Link data points: 12
- ^{b1} Total Number of PL-Link data points (under unconfigured status): 28
- ^{b2} Activate all selectable PL-Link devices, then check the maximum number of PL-Link data points: 9

22.4.12 Number of data points for DXR2

| ASN | Max. number of onboard IO- and KNX PL-Link data points | Description |
|--------------|--|--|
| DXR2.x09 | 30 | 1 DI, 2 UI, 3 AO, 3 Relay |
| DXR2.x09T | 30 | 1 DI, 2 UI, 4 Triac, 1 AO, 1 Relay |
| DXR2.x10 | 30 | 1 DI, 2 UI, 4 Triac, 3 Relay |
| DXR.x10PL | 30 | 1 Pressure, 1 DI, 2 UI, 4 Triac, 1 AO |
| DXR.x10PLX | 60 | 1 Pressure, 1 DI, 2 UI, 4 Triac, 1 AO |
| DXR2.x11 | 30 | 1 DI, 2 UI, 6 Triac, 2 AO |
| DXR2.x12P | 30 | 1 Pressure, 1 DI, 2 UI, 6 Triac, 2 AO |
| DXR2.x12PX | 60 | 1 Pressure, 1 DI, 2 UI, 6 Triac, 2 AO |
| DXR2.E17C-1 | 30 | 3 DI, 2 10K input impedance, 4 UI, 4 Triac, 4 AO |
| DXR2.E17CX-1 | 60 | 3 DI, 2 10K input impedance, 4 UI, 4 Triac, 4 AO |
| DXR2.x18 | 60 | 2 DI, 4 UI, 8 Triac, 4 AO |

Web clients for room operation

| Item | Limit | Description |
|--|--------|--|
| Standard web clients ¹ | 3 | Recommended number of web clients that can simultaneously access a DXR2. |
| Templates with standard background pictures ² | 2 | Maximum number of different templates which are using the default background pictures. |
| Customized background pictures ² | 1.5 MB | Maximum total size of all customized background pictures (the PNG file format is used as a reference). |

Key

- ¹ Restriction: When using standard web clients (web browser on PCs, smart phones, tablets, etc.), the screen display and operation (touch or mouse) are neither modified nor tested for the available browsers.
- ² Valid values when using 8 room applications at the boundary of maximum system limits.

22.4.13 PXM20 operator unit

| Item | Limit | Description |
|----------------------------------|-------|--|
| PX (no PXC3) | 50 | Number of PX that can be operated. The visibility of the PX automation stations can be limited on the BACnet network. This is only useful if the site is restricted to one BACnet network. For hardware series A devices (1 MB memory), the number of PX automation stations per site should be limited to 30. |
| Alarm administration | | Only the alarms from the site where the user is logged on are displayed (PXM20 self-registers as temporary alarm recipient for all devices of a site). |
| BACnet objects in alarm per site | 50* | Maximum number of BACnet objects per site. The administration of the number of BACnet objects in alarm per site is limited. Others cannot be displayed or operated in Alarm Viewer when there are more BACnet objects in alarm. |
| Alarm History | 50* | Maximum number of entries in the Alarm History. The oldest entries are deleted when this limit is exceeded. |

22.4.14 PXM10 operator unit

| Item | Limit | Description |
|---------------------------------|-------|--|
| PX (no PXC3) | 1* | Only the connected automation station / system controller can be operated. |
| Alarm administration | | Management of the alarms of the PXC to which the PXM10 is connected. |
| BACnet objects in alarm per PXC | 25* | Max. number of BACnet objects in alarm per PXC. The management of the number of BACnet objects in alarm per PXC is limited. Others cannot be displayed or operated in Alarm Viewer when there are more BACnet objects in alarm. |

22.4.15 Design Control Point

22.4.15.1 Device-related limits

| Function | Touch panel BACnet/IP | | BACnet/IP web interface | |
|--------------------------------------|---|--------------------|-------------------------|-------------|
| | PXM30.E | PXM40.E PXM50.E | PXG3.W100-2 | PXG3.W200-2 |
| Object view | All data points of all assigned devices | | | |
| Graphical operation (BACnet objects) | 500 | 1,000 | 1,000 | 2,000 |
| Haystack interface (BACnet objects) | 500 | 1,000 | 1,000 | 2,000 |
| Online trends | 20 | 20 | 20 | 50 |
| Graphics (average complexity) | 20 | 20 | 20 | 50 |

22.4.15.2 Memory management

| | PXM30.E | PXM40.E PXM50.E | PXG3.W100-2 | PXG3.W200-2 |
|--|---------------------------|---------------------------|---------------------------|----------------------------|
| Available application memory | 3 GB | 3 GB | 3 GB | 3 GB |
| Memory requirements | | | | |
| Max. number of DPs that can be integrated for graphical operation | 1,500 kB (500 DP) | 2,500 kB (1,000 DP) | 2,500 kB (1,000 DP) | 5,000 kB (2,000 DP) |
| Max. number of plant graphics at 300 kB each | 6,000 kB (20 graphics) | 6,000 kB (20 graphics) | 6,000 kB (20 graphics) | 15,000 kB (50 graphics) |
| Alarm history - 10,000 entries | 5,000 kB | 5,000 kB | 5,000 kB | 5,000 kB |
| 1 online trend (multistage object) 1 year, 24 entries per day = approx.. 9,000 values | 800 kB | 800 kB | 800 kB | 800 kB |
| 1 report with 1,000 objects | 200 kB | 200 kB | 200 kB | 200 kB |
| Documents function – max total size | -- | -- | -- | 50 MB |
| Documents function – max single document size | -- | -- | -- | 10 MB |
| Documents function – max documents | -- | -- | -- | 50 |

22.4.15.3 Graphics-based operation

Data point capacity is limited for graphical operation (data point integration).

The figures below apply to integration over the standard user level and where full graphical operation of all data points is required.

| Access level for integration | Desigo primary plant | | Desigo room automation | |
|------------------------------|-----------------------------------|----------------------------|---------------------------------------|--|
| | Extended operation | | Standard operation | |
| | Max. number * | Number of hardware-I/Os ** | Number of simple rooms (HVAC only) | Number of complex rooms (HVAC, lighting, shading) |
| PXG3.W100-2 | 400 | 30 | 10 | |
| PXG3.W200-2 | 800 | 60 | 20 | |
| PXM30.E | 200 | 15 | 5 | |
| PXM40.E | 400 | 30 | 10 | |
| PXM50.E | 400 | 30 | 10 | |
| PXM30-1, PXM40-1, PXM50-1 | n.a. (see limits for PXG3.Wx00-2) | | | |

* The selection of integrated data points can be optimized and individually customized with the Advanced Tool in the data point integration function.

** Primary plants: Per hardware I/O some 2.5 BACnet objects are integrated on average.

22.4.15.4 Technical limits

The following limits are recommended and verified.

| Function | Touch panel | | BACnet/IP web interface |
|---|-------------------------------|-------------------------------|----------------------------|
| | TCP/IP | BACnet/IP | |
| | PXM50-1 PXM40-1 PXM30-1 | PXM50.E PXM40.E PXM30.E | PXG3.W100-2 PXG3.W200-2 |
| Max. number of assigned devices BACnet/IP BACnet MS/TP *) | n.a. | 50 10 | 50 10 |

| Function | Touch panel | | BACnet/IP web interface |
|--|-------------------------------|-------------------------------|----------------------------|
| | TCP/IP | BACnet/IP | |
| | PXM50-1 PXM40-1 PXM30-1 | PXM50.E PXM40.E PXM30.E | PXG3.W100-2 PXG3.W200-2 |
| BACnet LonWorks *) | | 10 | 10 |
| Number of simultaneously connected operator clients ¹ | n.a. | 5 | 5 |
| Max number of devices connected to one BACnet device | n.a. | 5 | 5 |
| Data points per plant graphic | n.a. | 40 | 40 |
| Data points (in a table) | n.a. | 40 | 40 |
| Number of defined users | n.a. | 8 | 8 |
| Trend chart, max. number of entries (Max. 5 trends, with 10,000 each) | 50,000 | 50,000 | 50,000 |
| Max. entries per online trend ² | 10,000 | 10,000 | 10,000 |
| Max. entries per BACnet trendlog object ³ | 5000 | 5000 | 5000 |
| Trend export: Max. number of entries per trend | 10,000 | 10,000 | 10,000 |
| Max. alarm History entries | 10,000 | 10,000 | 10,000 |

*) see specific MSTP and LonWorks section for more specific limitations

¹ 5 is the recommended number. More clients can be connected, but performance deteriorates when the same operations are performed simultaneously. Example: Simultaneous load of the same plant graphic to 10 clients.

² Trend data management must be configured according as part of the online trend configuration.
10,000 trend entries can be recorded from:

- 1 trend entry daily over a minimum of 25 years
- 1 trend entry every hour over a minimum of 1 year
- 1 trend entry every 15 minutes over a minimum of 3 months

³ The time to read a BACnet trendlog object depends on the number of entries and the data link.

Desigo Control Point is optimized for BACnet/IP devices which are supporting COV communication. The objects of an assigned device will automatically be polled if the device does not support COV. Polling causes a higher network communication load. This can decrease the overall performance of the Desigo Control Point device, depending on the BACnet device and on the type of communication (MS/TP, LonWorks). Such topologies must be tested individually and project specifically.

22.4.15.5 Limits for MS/TP

Use the router PXG3.L / PXG3.M with **HW Index C only**.

Data point integration: MS/TP devices are considerably slower than IP devices due to the lower network speed. Reference value: Approximately 15 times slower (depending on the project setup).

MS/TP is tested with 10 devices that support COV (e.g. DXR2). Recommendation: Assign or integrate a maximum of 10 MS/TP devices on one network branch. More devices and/or network complexity are permitted, but they may result in longer and inconsistent times and degraded performance.

- Assign devices offline/online:
More than 10 devices degrade performance in ABT-SSA.
Additional network branches degrade performance in ABT-SSA.
- Device learning offline/online:
More than 10 devices takes longer.
The times are slower with additional network branches.
- Device restart (e.g., after the power returns):
More than 10 devices takes longer.
The times are slower for a restart.

The following limits and constraints must be considered if the MS/TP devices do not support COV, i.e. only polling:

- Disable the list view core function (user profile setting in ABT Site)
- Limit to one concurrent operating client (limit your network topology)
- Set baud rate to 38400 or higher
- Only use the online trends functionality very limited
- Limit the points per graphic to 15

22.4.15.6 LonWorks limits

Use the router PXG3.L / PXG3.M with **HW Index C only**.

Data point integration: LonWorks devices take a lot longer than IP devices due to the lower network speed. Reference value: Approximately 30 times longer (depending on the project setup).

To avoid timeout problems,

- Use the workflow: Data point integration.
- Integrate at the standard user level
- Avoid the "Cache" dialog
- Create templates for individual data point selection for LON devices on IP devices. These templates can then be imported to the LON devices.

22.4.16 PXG3.L and PXG3.M BACnet routers

| Item | Limit | Description |
|------------------------------------|--|---|
| BDT (Broadcast Distribution Table) | [50*] | Max. number of BBMDs (BACnet Broadcast Management Devices) in a BACnet internetwork. If a BACnet router is in its own IP segment, it must be configured as a BBMD. |
| FDT (Foreign Device Table) | [50*] | Maximum number of foreign devices which can register with the BACnet router. Desigo CC in a remote IP segment counts as a foreign device. |
| Ethernet bit rate | 10/100 Mbit/s | The router supports 10/100 Mbps. |
| MS/TP telegrams | [100 - 140] pkt/s @ 115,200 bps [-120] pkt/s @ 76,800 bps Max. [-4,5] KB/s | The BACnet router integrates BACnet MS/TP not as a field bus in the network. The router operates transparently and routes all data traffic addressed to the subnet. This is why global broadcast telegrams negatively impact transmission performance of the router and end devices. Recommendation: Do not carry out time and security-critical process controls using BACnet MS/TP. Depends on baudrate, number of nodes and maximum number of data frames (N max_info_frames). |
| BACnet/LonTalk | [100 - 120] pkt/s @ 78 KB/s Max. [-4,5] KB/s | The BACnet router integrates one (1) BACnet/LonTalk network. The router operates transparently. The same restrictions apply to global broadcast telegrams as for MS/TP. |
| BACnet/IPv4 | [~2500] pkt/s Max. [-500] KB/s | The BACnet router can route between two BACnet/IP networks. The BACnet/IP networks have different UDP ports. |
| BACnet/IPv6 | 1 | The BACnet router integrates one (1) BACnet/IPv6 network. The router works transparent, but when connection ports for BACnet/IPv4 and BACnet/IPv6 are used simultaneously, make sure that no unintentional ethernet loops are created on the IT side. |

22.4.17 SX OPC

| Item | Limit | Description |
|--|---------|--|
| SX OPC applications | 1 | SX OPC application per PC. The performance depends on the PC hardware. |
| OPC server | [10] | Max. number; OPC data access 2.x or 3.0 specification. |
| BACnet objects | 20,000* | Maximum number of BACnet objects. |
| Configured alarm recipients | 3* | |
| Temporary alarm receiver | 20* | Minus configured alarm recipients. |
| Alarm-generating objects | [2,000] | Alarm-generating objects (of total 20,000 BACnet objects). |
| SX BACnet references client resources ¹ | [1,000] | |
| Trendlog objects | 1000 | Maximum number. |
| Scheduler program / Scheduler objects | [15] | Per BACnet server. |
| Calendar objects | [10] | |

Key

- ¹ Max. number of BACnet client connections (COF or polling), that is, values read from or written to (commanded) the own automation station or a remote automation station.
- BACnet client connections are used in Input, Output, Scheduler, Trendlog and Group objects (all NameRef_Type inputs with AddrKind = B). The configured alarm receivers of the Notification Class objects do NOT require any BACnet client references.
- The available number of BACnet client references does not address more than 50 different remote automation stations.

22.4.18 Desigo CC

For the system configurations of the Desigo CC management platform, see *Desigo CC System Description* (A6V10415500).

22.4.19 Desigo Insight

For the system configurations of the Desigo Insight management station V6.0 SP2, see *Desigo Building Automation System 6.0 SP, Technical Principles* (CM110664 / 2016-09-20).

22.4.20 Desigo Xworks Plus (XWP)

| Item | Limit | Description |
|--|-------|---|
| Length of site name | 9 | Max. 9 characters. |
| Number of XWP per BACnet internetwork (parallel engineering) | 10 | Parallel engineering is possible under the following limitations: Node setup: Only one XWP per LonWorks/IP segment. Download and online operation: Only one XWP per automation station. |
| Number of I/O function block instance per plan | 200 | The number of I/O function block instances are limited per plan (compound). Mapping of function blocks on BACnet sets the limit. The limit is lower for other function blocks mapped to BACnet. |

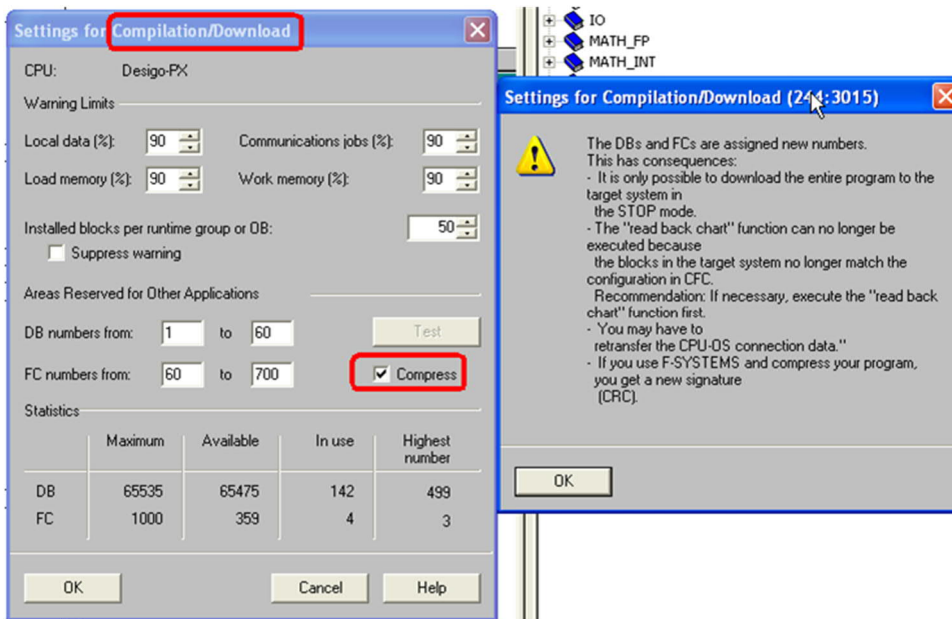
Problems with a high number of data points per automation station

When the maximum number of data points for a PXC..U is reached (350), it may no longer be possible to load the program into the PX automation stations due to the number of data blocks generated during compilation.

In this case, carry out the following steps on the PX automation station:

1. Reload parameters.
2. Run Reorganize in the **PX Design Manager**.
3. Go to **Tools > Settings > Compilation download** and select the **Compress** check box.

4. Recompile the data.
5. Perform a full download.



22.4.21 Desigo Automation Building Tool (ABT)

| Item | Limit | Description |
|-----------------|---------|--|
| Function blocks | [8,000] | Max. number of function blocks per application function. |

22.5 Applications

22.5.1 Peak Demand Limiting (PDL)

| Item | Limit | Comment |
|-----------------|-------|---|
| Monitored loads | [28*] | Max. number of monitored loads. |
| Tariff limits | 4* | Max. number of configurable tariff limits. |
| Cycle time [ms] | 500 | <p>Minimum cycle time required to ensure the functioning of the PDL application.</p> <p>To guarantee the cycle time, use a PX modular automation station (PXC 100/200...D, PXC12/22/36...D).</p> <p>Do not use the the automation station with the PDL application to control any other plant.</p> <p>The PDL application must be confined to one automation station only.</p> <p>Limit control is binary only (enabled/disabled). Step control (Stage1, Stage2, Stage3) or modulating control (0...100%) is not possible.</p> <p>Commissioning and operation are only possible with XWP.</p> <p>There is no provision for backward compatibility with future PDL applications.</p> |

23 Compatibility

For information on the system compatibility of the Desigo CC management platform, see *Desigo CC System Description* (A6V10415500).

For information on the system compatibility of the Desigo Insight management station V6.0 SP2, see *Desigo Building Automation System 6.0 SP, Technical Principles* (CM110664 / 2016-09-20).

For the current state of the Valid Version Set (VVS), see the document *Desigo_VVS_6.10.48x.pdf*.

Glossary

| Term | Description |
|-------------------------|--|
| Project data | Desigo engineering and project data required to create runtime systems, but that are no longer needed for operation (offline data). |
| Runtime system | Firmware (loaded) installed on the hardware of the customer plant or software with compiled project data including libraries (online data). |
| New | New Desigo customer project with no Desigo runtime system and project data. |
| Extension | Existing plant or installation (existing Desigo runtime system with project data) that is being expanded or extended (e.g., additional buildings). |
| Migration | Replacement of existing plant or installation (existing Desigo / Visonik / Unigr / Integral runtime system with project data) by new technology with a change of software and/or hardware. |
| Upgrade | Functional improvement to existing plant or installation (existing Desigo runtime system with project data) by deploying developments for a new Desigo system version. |
| Update | Existing plant or installation (existing Desigo runtime system with project data) is updated within the same version (e.g., to eliminate errors with a service pack). |
| Project data conversion | <p>Online Desigo project data from earlier Desigo versions are migrated to the current ABT/XWP version when opened in ABT/XWP.</p> <p>During conversion, the existing database structure and/or associated tool landscape is migrated to the latest version. A conversion always impacts all project data of a tool project.</p> <p>The project data and libraries remain unchanged. The runtime system (online project data) does not change, that is, the original version status remains as is.</p> |

23.1 Desigo version compatibility definition

General definition

The Desigo version compatibility describes the compatibility of Desigo products:

- Within a Desigo Xworks Plus (XWP) project (incl. ABT/SSA)
- With the same tool project data
- On a Desigo runtime system

The compatibility also comprises Desigo project data at both the management level and room automation linked to the same Desigo Xworks Plus (XWP) project.

Desigo system versions

The term refers to the various development phases of the Desigo building automation and control system. The currently supported versions are:

- Desigo V2.2
- Desigo V2.3
- Desigo V2.35
- Desigo V2.36
- Desigo V2.37
- Desigo V4.0
- Desigo V4.1
- Desigo V5.0

- Desigo V5.1
- Desigo V6.0
- Desigo V6.1
- Desigo V6.2
- Desigo V6.2 Update
- Desigo V6.2 Update 2
- Desigo V6.2 Update 3
- Desigo V6.2 Update 4

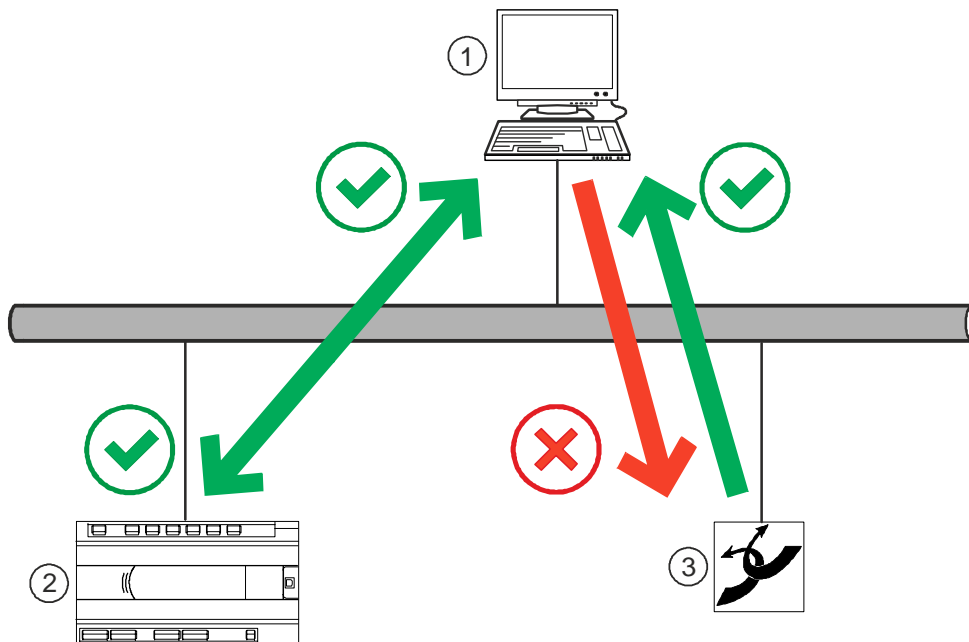
23.2 Desigo system compatibility basics

23.2.1 Compatibility with BACnet standard

Desigo supports the following BACnet protocol revisions:

- Desigo CC: 1.15
- Desigo room automation stations: 1.13
- Desigo PX, PXM20: 1.12
- Desigo Control Point PXG3.Wxxx-1 and PXMx0: 1.13
- PXG3 router: 1.13

There are third-party BACnet devices on the market that support higher BACnet protocol revisions.



Key

- ① Desigo CC
- ② Desigo devices with a lower revision or the same revision as Desigo CC
- ③ Third-party BACnet devices with a higher revision than Desigo CC
- ✓ Properties from earlier BACnet protocol revisions can be read by a BACnet device even if the device supports a higher BACnet protocol revision than Desigo CC.
- ✗ New properties from a BACnet protocol revision higher than Desigo CC cannot be read or changed, because they are not recognized by Desigo CC.

The BACnet client ensures the backwards compatibility. Desigo CC should thus have a BACnet revision that is at least the same as all of its connected BACnet servers.

Usually, BACnet devices of a specific BACnet protocol revision fully support earlier revision functions. However, since this is not true in all cases, we recommend that you verify the compatibility in each case.

For an overview of the BACnet functions supported in Desigo, see *BACnet Protocol Implementation Conformance Statement (PICS)* (CM110665).

Create and delete BACnet objects

The function can be used on Desigo PXC automation stations. Desigo room automation stations PXC3 are not supported.

Third-party devices can be processed using the same functionality as long as they support creating and deleting BACnet objects. PXM20 operating units do not display dynamic objects.

Backup and restore BACnet devices

With the BACnet backup and restore function you can upload saved program data (application program) from a BACnet device to Desigo CC and restore it to the same or a new BACnet device.

The backup and restore function can only be run if the third-party BACnet devices support it.

Compatibility Desigo Insight

For information on the system compatibility of the Desigo Insight management station V6.0 SP2, see *Desigo Building Automation System 6.0 SP, Technical Principles* (CM110664 / 2016-09-20).

Compatibility Desigo CC

For information on the system compatibility of the Desigo CC management platform, see *Desigo CC System Description* (A6V10415500).

23.2.2 Compatibility with operating systems

Microsoft client operating systems

| Tool | Compatibility with Microsoft client operating systems |
|-----------------------------------|---|
| | Windows 10 Professional / Enterprise |
| | 64-bit |
| XWP | Yes* |
| ABT Site / ABT Pro | Yes* |
| RXT10.5 | Yes* |
| Desigo Configuration Module (DCM) | Yes* |

Status April 2020:

* tested with release versions (1809), 1903, 1909.

Version in parentheses () for DWP client (Siemens internal) only.

Note

Windows 7 and Windows 8.1 are not supported.

For more information about certificates, see *IT Security in Desigo Installations* (CM110663).

Unlisted Microsoft client operating systems/editions (especially Home Premium or 32-bit versions) are not supported.

Branch Office Server (BOS) only supports server operating systems.

LMS/LMU supports several Microsoft client operating systems. For details on requirements and limitations, see *License Management Utility* (A6V10455206).

Microsoft server operating systems

The end user is responsible for the correct licensing of any third-party licenses.

| Product | Compatibility with Microsoft server operating systems | | |
|----------------------------|---|------------------------------|------------------------------|
| | Windows Server 2012 R2 Standard | Windows Server 2016 Standard | Windows Server 2019 Standard |
| | 64-bit | 64-bit | 64-bit |
| Branch Office Server (BOS) | Yes | Yes | Yes |

Unlisted Microsoft server operating systems/editions are not supported. They can, however, be used for stand-alone SQL servers and file hosts.

Note

Windows Server 2012 and 2012 R2 support

Microsoft has officially announced the end of support for Windows Server 2012 and 2012 R2 for October 10th 2023. Therefore, tools on Windows Server 2012 and 2012 R2 will no longer be supported after that date. We recommend using Windows Server 2016 or any of the other compatible operating systems instead.

23.2.3 Compatibility with SQL servers

| Product | Compatibility with Microsoft SQL servers | | | | |
|----------------------------|--|-----------------|-----------------|-----------------|-----------------|
| | SQL server 2012 | SQL server 2014 | SQL server 2016 | SQL server 2017 | SQL server 2019 |
| | Standard | Standard | Standard | Standard | Standard |
| | 64-bit | 64-bit | 64-bit | 64-bit | 64-bit |
| Branch Office Server (BOS) | Yes | Yes | Yes | Yes | Yes |

Unlisted SQL server versions/editions are not supported.

The Branch Office Server (BOS) is compatible with the following operating systems:

- SQL Server 2012/2014 Standard on Windows Server 2012 R2 Standard Edition
- SQL Server 2016 Standard on Windows Server 2016 Standard Edition
- SQL Server 2017 Standard on Windows Server 2016 Standard Edition
- SQL Server 2019 Standard on Windows Server 2019 Standard Edition

23.2.4 Compatibility with Microsoft Office

| Product | Version | Microsoft Office |
|--|---------|---|
| ABT Site / ABT Pro | V4.3 | MS Office 2019 (64-bit edition) MS Office 365 ProPlus (64-bit edition) |
| Desigo Xworks Plus (XWP) incl. auxiliary tools | V6.3 | |
| Desigo Configuration Module (DCM) | V6.3 | |

23.2.5 Compatibility with web browsers

| Product | Tested compatibility with web browsers | | | |
|----------------------------------|--|---------------|---------|-----------|
| | Microsoft Edge as of 80.0.361 (Chromium) | Google Chrome | Firefox | Safari 10 |
| SSA ¹ / Web Interface | Yes | Yes | Yes | Yes |
| ABT Site online help | Yes | Yes | Yes | No |
| Desigo Control Point | See chapter <i>Desigo Control Point</i> | | | |

Key

¹ Support of HTML5-capable browsers with native SVG format.

Desigo CC

For notes on Desigo CC web client running in a browser shell, see *Desigo CC System Description* (A6V10415500).

23.2.6 Compatibility with ABT Go

ABT Go is a mobile tool for commissioning and maintenance tasks of Siemens devices used in building automation and control systems.

| Operating system | Version |
|------------------|------------------------------|
| Android | Version 4.4 KitKat and above |
| iOS* | iOS 10 and above |

* Do not minimize ABT Go and make sure the device remains unlocked for longer running tasks such as generating a report.

For Android devices download ABT Go from the Google Play Store:

<https://play.google.com/store/apps/details?id=com.siemens.abtgo&hl=en>

For iOS devices download ABT Go from the Apple Store:

<https://itunes.apple.com/app/abt-go/id1293043551?l=en&ls=1&mt=8>

23.2.7 Compatibility with VMware (virtual infrastructure)

| Product | Version | VMware version |
|---|---------|-----------------------|
| Desigo Xworks Plus (XWP), ABT/SSA, and other additional tools | V6.3 | VMware Workstation 15 |

23.2.8 Compatibility of software/libraries on the same PC

All Desigo software and LibSets (standard application libraries) must be on the same computer and must have the same version.

Mixing different versions of PX libraries on devices is not allowed within the same application.

You can install Desigo CC, RXT10.5 (if required), Desigo Xworks Plus (XWP) and ABT on one PC in any order. But you must install the libraries at the end.

Installing Desigo software from different systems versions on the same PC is not supported. Check operating system compatibility prior to installing.

23.2.9 Hardware and firmware compatibility

Desigo hardware and firmware is only partially compatible to products from earlier versions in the same Desigo project runtime system.

BACnet peer-to-peer communication between Desigo PX devices from the earliest version to the current version is guaranteed.

Restrictions

As soon as an automation station or a system controller with the newest Desigo firmware is used in a runtime system, all operating clients and Desigo Control Point must be upgraded to the same version. Otherwise, only limited operation is available.

Not all Desigo PX devices in the field can be upgraded to the newest firmware version.

For information on the hardware and firmware compatibility of the Desigo CC management platform, see *Desigo CC System Description* (A6V1041550).

TX-I/O

A firmware update or upgrade from TX-I/O modules is not possible (except for TXI1.OPEN, TXI2.OPEN, and TXI2-S.OPEN).

To load firmware, protocol applications, and the configuration to the TX Open modules, use the TX Open Tool, which is available as a part of XWP.

See *Desigo Xworks Plus Online Help* (CM111006).

23.2.10 Backward compatibility

Desigo software and libraries are downwards compatible. Desigo products can process data compiled with earlier versions.

Restrictions

After upgrading Desigo project data for a Desigo software product to the newest version, the data can only be accessed or processed with the appropriate software/LibSets for the newest version.

23.2.11 Engineering compatibility

All project data on all system levels must have the same LibSet with the same LibSet version number for unlimited engineering of tested Desigo solutions (libraries).

To use the current library extensions, all project data must be upgraded to the current version.

When engineering a tool project, all tool installations must have the same version as the project.

23.2.12 Compatibility with Desigo Configuration Module (DCM)

The Desigo Configuration Module (DCM) version supplied with the relevant Desigo system covers the available Desigo product scope.

When importing DCM projects from earlier DCM versions, the project is converted to the status and options of the current DCM version. After conversion, the project data can only be revised in the current DCM version.

DCM projects from DCM version 5.0 support import.

DCM requires Microsoft Office.

Note

Automation stations PXC4 and PXC5, as well as the DXR1 range are not supported in DCM.

23.2.13 Compatibility with Desigo PX / Desigo room automation

The modular Desigo PX automation stations / system controllers PXC00/50/100/200-E.D are supported as Desigo room automation system function controllers for the PXC3 room automation stations. For performance reasons, use PXC00-E.D where possible.

The local operator unit PXM10 cannot be used together with the following devices:

- PXC3 room automation stations
- PX KNX (PXC001.D/PXC001-E.D)
- PXG3.L/PXG3.M (BACnet router)
- PXG3.Wxxx (web interface BACnet/IP of Desigo Control Point)

No I/O modules may be connected to the system controller LonWorks PXC00(-E).D.

The modular series PXC...D (Desigo PX) and the PXC3 room automation stations do not have a PPS2 connection.

23.2.14 Compatibility with Desigo RX tool

RXT10.5 supports only system integration via the PXX L11/L12 Controller.

You can only use LON standard tools NL220 (Newron System) or LonMaker (Echelon) with project data from LonWorks PXC00(-E).D or PXC50/100/200 (-E).D system controllers.

RXT10.x project data

All Desigo software and LibSets must be on the same PC and have the same system version.

23.2.15 Compatibility with TX-I/O

| Product range | TX-I/O modules | | | | | | | | | | | |
|--|----------------|--------------|-------------|----------------|-------------|----------------|-------------|---------------|-------------|----------------|--------------|-------------|
| | TXM1. 8D | TXM1. 16D | TXM1. 8U | TXM1. 8U-ML | TXM1. 8X | TXM1. 8X-ML | TXM1. 6R | TXM1. 6R-M | TXM1. 8P | TXM1. 6RL | TXM1. 8RB | TXM1. 8T |
| Modular room automation stations PXC3 (from index D) | • | • | • | - | - | - | • | - | - | • | • | • |
| Modular room automation stations PXC..D | • | • | • | • | • | • | • | • | • | • ¹ | - | • |

Key

- ¹ Directly switched lighting applications (by the user) are not supported by the PXC..D automation stations. For this reason, the configured button function of the digital input modules is not available together with the PXC...D automation stations.

23.2.16 Compatibility with TX Open

The TXI1.OPEN, TXI2.OPEN, and TXI2-S.OPEN TX Open modules can only be used together with the PXC50/100/200(-E).D and PXC22.1/PXC36.1(-E).D automation stations.

23.2.17 KNX PL-Link devices in ABT Site and ABT Pro

| | | | | | | Supported workflows | | | | | |
|-------------------|----------------|---------------|---|---------------------------|-------------|--|--|---|----------------------------|---|----------------------------|
| | | | | | | Room automation (without Critical Environment) | | | Critical Environment | | |
| | | | | | | ABT Pro version or higher | | | ABT Site version or higher | ABT Pro version or higher | ABT Site version or higher |
| Product portfolio | Product group | ASN | Description | PL-Link power consumption | Data points | Hardware catalog | As one instance in application library | Fully integrated in application library | In application types | Fully integrated in application library | Application types |
| Acvatix | Valve actuator | GDB111.9E/KN | 6-way valve actuator (5 Nm non-spring return) | 5 mA | 1 | V4.0 | V4.0 | V4.0 | V4.0 | No | No |
| Acvatix | Valve actuator | GLB111.9E/KN | 6-way valve actuator (10 Nm non-spring return) | 5 mA | 1 | V4.0 | V4.1 | No | No | No | No |
| Acvatix | Valve actuator | SSA118.09HK N | Room valve actuator | 10 mA | 1 | V4.0 | V4.0 | No | No | No | No |
| Desigo | Gateway | RXZ97.1 | KNX EnOcean Gateway | N/A | N/A | No | No | No | No | No | No |
| Desigo | I/O block | RXM21.1 | PL-Link I/O blocks for fan coil units | 5 mA | 5 | V4.0 | V4.0 | V4.0 | No | No | No |
| Desigo | I/O block | RXM39.1 | PL-Link I/O blocks for fan coil units | 5 mA | 5 | V4.0 | V4.0 | V4.0 | V4.0 | No | No |
| Desigo | Room unit | QMX2.P33 | QMX2 wall with display & T: without Green Leaf and KNX S-Mode | 10 mA | 3 | V4.0 | V4.0 | V4.0 | V4.0 | No | V4.1 |
| Desigo | Room unit | QMX2.P43 | QMX2 wall with display & T, rH: without Green Leaf and KNX S- | 10 mA | 4 | V4.0 | V4.0 | V4.0 | V4.0 | No | V4.1 |

| | | | | | | Supported workflows | | | | | |
|-------------------|---------------|-----------|--|---------------------------|-------------|--|--|---|----------------------------|---|----------------------------|
| | | | | | | Room automation (without Critical Environment) | | | | Critical Environment | |
| | | | | | | ABT Pro version or higher | | | ABT Site version or higher | ABT Pro version or higher | ABT Site version or higher |
| Product portfolio | Product group | ASN | Description | PL-Link power consumption | Data points | Hardware catalog | As one instance in application library | Fully integrated in application library | In application types | Fully integrated in application library | Application types |
| | | | Mode | | | | | | | | |
| Desigo | Room unit | QMX3.P02 | QMX3 wall with switches | 8 mA | 5 | V4.0 | V4.0 | V4.0 | V4.0 | No | No |
| Desigo | Room unit | QMX3.P30 | Freely configurable operator unit, wall-mounted | 8 mA | 1 | V4.0 | V4.0 | V4.0 | V4.0 | No | No |
| Desigo | Room unit | QMX3.P34 | Freely configurable operator unit, wall-mounted | 8 mA | 3 | V4.0 | V4.0 | V4.0 | V4.0 | No | No |
| Desigo | Room unit | QMX3.P35H | QMX3 wall with touch for HVAC | 15 mA | 3 | V4.0 | V4.0 | No | No | No | No |
| Desigo | Room unit | QMX3.P36 | Freely configurable flush-mounted room unit | 13 mA | 3 | V4.0 | V4.0 | V4.0 | V4.0 | No | No |
| Desigo | Room unit | QMX3.P37 | Freely configurable operator unit, wall-mounted | 10 mA | 7 | V4.0 | V4.0 | V4.0 | V4.0 | No | No |
| Desigo | Room unit | QMX3.P38H | QMX3 wall with touch for HVAC, lighting, shading | 15 mA | 7 | V4.0 | V4.1 | No | No | No | No |
| Desigo | Room unit | QMX3.P40 | Room operator unit without display, with temperature and humidity sensor | 8 mA | 2 | V4.0 | V4.0 | V4.0 | V4.0 | No | No |

| | | | | | | Supported workflows | | | | | |
|-------------------|----------------|-----------|--|---------------------------|-------------|--|--|---|----------------------------|---|----------------------------|
| | | | | | | Room automation (without Critical Environment) | | | Critical Environment | | |
| | | | | | | ABT Pro version or higher | | | ABT Site version or higher | ABT Pro version or higher | ABT Site version or higher |
| Product portfolio | Product group | ASN | Description | PL-Link power consumption | Data points | Hardware catalog | As one instance in application library | Fully integrated in application library | In application types | Fully integrated in application library | Application types |
| Desigo | Room unit | QMX3.P44 | QMX3 wall with display & T, rH | 10 mA | 4 | V4.1 | V4.1 | No | No | No | No |
| Desigo | Room unit | QMX3.P70 | Freely configurable operator unit, wall-mounted | 15 mA | 3 | V4.0 | V4.0 | V4.0 | V4.0 | No | No |
| Desigo | Room unit | QMX3.P74 | Freely configurable operator unit, wall-mounted | 15 mA | 5 | V4.0 | V4.0 | V4.0 | V4.0 | No | No |
| Desigo | Room unit | QMX3.P87 | Freely configurable operator unit, wall-mounted (fume hood) | 8 mA | 3 | V4.0 | V4.0 | N/A | N/A | V4.0 | V4.0 |
| Desigo | Room unit | QMX3.P88 | Freely configurable operator unit, flush-mounted (fume hood) | 8 mA | 3 | V4.0 | V4.0 | N/A | N/A | V4.0 | V4.0 |
| Gamma | Blind actuator | JB 520C23 | 1x blind actuator, 1x6A (AC 120 V) | 10 mA | 1 | V4.0 | V4.0 | V4.0 | V4.0 | No | No |
| Gamma | Blind actuator | JB 521C23 | 2x blind actuator, 2x6A (AC 120 V) | 10 mA | 4 | V4.0 | V4.0 | V4.0 | V4.0 | No | No |
| Gamma | Blind actuator | N 543D31 | Blind actuator: 4 channels | N/A | 4 | V4.1 | V4.1 | No | No | No | No |
| Gamma | Blind actuator | N 543D51 | Blind actuator with end position detection: 8 | N/A | 8 | V4.0 | V4.1 | No | No | No | No |

| | | | | | | Supported workflows | | | | | |
|-------------------|--------------------|-----------|--|---------------------------|-------------|--|--|---|----------------------------|---|----------------------------|
| | | | | | | Room automation (without Critical Environment) | | | | Critical Environment | |
| | | | | | | ABT Pro version or higher | | | ABT Site version or higher | ABT Pro version or higher | ABT Site version or higher |
| Product portfolio | Product group | ASN | Description | PL-Link power consumption | Data points | Hardware catalog | As one instance in application library | Fully integrated in application library | In application types | Fully integrated in application library | Application types |
| | | | channels | | | | | | | | |
| Gamma | Blind actuator | RL 521/23 | 2 x blind actuator, 6A, AC 230V | 10 mA | 4 | V4.0 | V4.0 | V4.0 | V4.0 | No | No |
| Gamma | Blind actuator | RS 520/23 | 1 x blind actuator, 6A, AC 230V | 10 mA | 2 | V4.0 | V4.0 | V4.0 | V4.0 | No | No |
| Gamma | Blind actuator | UP 520/03 | 1x blind actuator, 6A, AC 230V | 10 mA | 2 | V4.0 | V4.0 | V4.0 | V4.0 | No | No |
| Gamma | Blind actuator | UP 520/13 | 1 x blind actuator, 6A, AC 230V | 10 mA | 2 | V4.0 | V4.0 | V4.0 | V4.0 | No | No |
| Gamma | Input | JB 260C23 | 4-fold binary input, AC/DC12...230 V | N/A | N/A | V4.0 | V4.0 | V4.0 | V4.0 | No | No |
| Gamma | Input | RL 260/23 | 4-fold binary input, AC/DC12...230 V | N/A | 4 | V4.0 | V4.0 | V4.0 | V4.0 | No | No |
| Gamma | Light dim actuator | JB 525C23 | 1x universal lighting dimmer, 1x125VA (AC120V) | 10 mA | 1 | V4.0 | V4.0 | V4.0 | V4.0 | No | No |
| Gamma | Light dim actuator | JB 526C23 | 2 channel 1...10V dimmer with relay output | 10 mA | 2 | V4.0 | V4.0 | V4.0 | V4.0 | No | No |
| Gamma | Light dim actuator | JB 527C23 | 1 channel 1...10V | 10 mA | 1 | V4.0 | V4.0 | V4.0 | V4.0 | No | No |

| | | | | | | Supported workflows | | | | | |
|-------------------|--------------------|----------|--|---------------------------|-------------|--|--|---|----------------------------|---|----------------------------|
| | | | | | | Room automation (without Critical Environment) | | | Critical Environment | | |
| | | | | | | ABT Pro version or higher | | | ABT Site version or higher | ABT Pro version or higher | ABT Site version or higher |
| Product portfolio | Product group | ASN | Description | PL-Link power consumption | Data points | Hardware catalog | As one instance in application library | Fully integrated in application library | In application types | Fully integrated in application library | Application types |
| | | | dimmer with relay output | | | | | | | | |
| Gamma | Light dim actuator | N 525D11 | Switching/Dimming actuator, 2x DALI, broadcast | 5 mA | N/A | V4.1.1 | V4.1.1 | No | No | No | No |
| Gamma | Light dim actuator | N 528C01 | DIN rail 2-channel universal lighting dimmer (incl. LED dimming), UL | N/A | N/A | V4.0 | V4.0 | V4.0 | V4.0 | No | No |
| Gamma | Light dim actuator | N 528D01 | Universal dimmer, 2 x 300 VA, AC 230 V | N/A | 2 | V4.0 | V4.0 | V4.0 | V4.0 | No | No |
| Gamma | Light dim actuator | N 536D31 | Switching/Dimming actuator 4 channel, 1...10V dimmer with relay output | N/A | 4 | V4.1.1 | V4.1.1 | No | No | No | No |
| Gamma | Light dim actuator | N 536D51 | Switching/Dimming actuator 8 channel, 1...10V dimmer with relay output | N/A | 8 | V4.0 | V4.1.1 | No | No | No | No |
| Gamma | Light dim actuator | N 554D31 | Universal dimmer, 4 x 300 VA / 1x 1000 VA, AC 230 V | 7.5 mA | 4 | V3.2 | V4.0 | V4.0 | V4.0 | No | No |

| | | | | | | Supported workflows | | | | | |
|-------------------|-----------------------|-----------|--|---------------------------|-------------|--|--|---|----------------------------|---|----------------------------|
| | | | | | | Room automation (without Critical Environment) | | | Critical Environment | | |
| | | | | | | ABT Pro version or higher | | | ABT Site version or higher | ABT Pro version or higher | ABT Site version or higher |
| Product portfolio | Product group | ASN | Description | PL-Link power consumption | Data points | Hardware catalog | As one instance in application library | Fully integrated in application library | In application types | Fully integrated in application library | Application types |
| Gamma | Light dim actuator | RL 526D23 | Switching/Dimming actuator 2 x AC 230 V, 10 A, 1...10 V | N/A | 2 | V4.1.1 | V4.1.1 | No | No | No | No |
| Gamma | Light dim actuator | RS 525/23 | 1 x universal lighting dimmer, 1x 250VA, AC 230V | 10 mA | 1 | V4.0 | V4.0 | V4.0 | V4.0 | No | No |
| Gamma | Light dim actuator | UP 525/03 | Universal lighting dimmer, 1x 250VA, AC 230V | 10 mA | 1 | V4.0 | V4.0 | V4.0 | V4.0 | No | No |
| Gamma | Light dim actuator | UP 525/13 | 1x universal light. dimmer, 1x 250VA, AC 230V | 10 mA | 1 | V4.0 | V4.0 | V4.0 | V4.0 | No | No |
| Gamma | Light switch | JB 510C23 | 2x lights switching, 2x10A (AC 120V/AC 277V) | 10 mA | 2 | V4.0 | V4.0 | V4.0 | V4.0 | No | No |
| Gamma | Light switch actuator | JB 512C23 | 1x light switching actuator, 1x20A or 1x15A (AC 120V / AC 277V or AC 347V) | 10 mA | 1 | V4.0 | V4.0 | V4.0 | V4.0 | No | No |
| Gamma | Light switch | JB 513C23 | 3x lights switching, 3x6A (AC | 10 mA | 3 | V4.0 | V4.0 | V4.0 | V4.0 | No | No |

| | | | | | | Supported workflows | | | | | |
|-------------------|---------------|-----------|--|---------------------------|-------------|--|--|---|----------------------------|---|----------------------------|
| | | | | | | Room automation (without Critical Environment) | | | Critical Environment | | |
| | | | | | | ABT Pro version or higher | | | ABT Site version or higher | ABT Pro version or higher | ABT Site version or higher |
| Product portfolio | Product group | ASN | Description | PL-Link power consumption | Data points | Hardware catalog | As one instance in application library | Fully integrated in application library | In application types | Fully integrated in application library | Application types |
| | | | 120V / AC 277V) | | | | | | | | |
| Gamma | Light switch | RL 512/23 | 1x light switching actuator, 16A / AC 230V | 10 mA | 1 | V4.0 | V4.0 | V4.0 | V4.0 | No | No |
| Gamma | Light switch | RL 513/23 | 3x lights switching actuator, 3x6A AC 230V | 10 mA | 3 | V4.0 | V4.0 | V4.0 | V4.0 | No | No |
| Gamma | Light switch | RS 510/23 | 2x lights switching actuator, 2x10A, AC 230V | 10 mA | 2 | V4.0 | V4.0 | V4.0 | V4.0 | No | No |
| Gamma | Light switch | UP 510/03 | 2x lights switching actuator, 2x10A, AC 230V | 10 mA | 2 | V4.0 | V4.0 | V4.0 | V4.0 | No | No |
| Gamma | Light switch | UP 510/13 | 2x lights switching actuator, 2x10A, AC 230V | 10 mA | 2 | V4.0 | V4.0 | V4.0 | V4.0 | No | No |
| Gamma | Push button | UP 20x | Arina push button (CN market only): UP 201 (single), UP 202 (double), UP 203 (quadruple) | 9 mA | N/A | No | No | No | No | No | No |

| | | | | | | Supported workflows | | | | | |
|-------------------|---------------|-----------|---|--|-------------|--|--|---|----------------------------|---|---------------------------------------|
| | | | | | | Room automation (without Critical Environment) | | | Critical Environment | | |
| | | | | | | ABT Pro version or higher | | | ABT Site version or higher | ABT Pro version or higher | ABT Site version or higher |
| Product portfolio | Product group | ASN | Description | PL-Link power consumption | Data points | Hardware catalog | As one instance in application library | Fully integrated in application library | In application types | Fully integrated in application library | Application types |
| Gamma | Push button | UP 21x | Touch sensors glass: UP 211 (single), UP 212 (double), UP 213 (quadruple) | 211: 15 mA 212: 20 mA 213: 25 mA | 2, 4, 8 | V4.0 | V4.0 | No | No | No | No |
| Gamma | Push button | UP 220D31 | Push-button interface: for multi vendor switches (4 DI or 4 DO for LED) | 11 mA | 4 | V4.0 | V4.0 | V4.0 | V4.0 | No | No |
| Gamma | Push button | UP 22x | Delta i-system push button: UP 221 (single), UP 222 (double), UP 223 (triple) | 9 mA | 2, 4, 6 | V4.0 | V4.0 | V4.0 | V4.0 | No | Partially from Desigo Room Automation |
| Gamma | Push button | UP 28x | Delta style push button: UP 285 (single), UP 286 (double), UP 287 (quadruple) | 9 mA | 2, 4, 8 | V4.0 | V4.0 | V4.0 | V4.0 | No | Partially from Desigo Room Automation |
| Gamma | Sensor | UP 255D21 | Brightness sensor | 10 mA | 1 | V4.0 | V4.0 | V4.0 | V4.0 | No | No |
| Gamma | Sensor | UP 258D11 | Presence detector & brightness sensor (product phase out at the end of 2016) | 10 mA | N/A | V4.0 | V4.0 | V4.0 | V4.0 | No | No |

| | | | | | | Supported workflows | | | | | |
|-------------------|-----------------|-----------|--|---------------------------|-------------|--|--|---|----------------------------|---|----------------------------|
| | | | | | | Room automation (without Critical Environment) | | | | Critical Environment | |
| | | | | | | ABT Pro version or higher | | | ABT Site version or higher | ABT Pro version or higher | ABT Site version or higher |
| Product portfolio | Product group | ASN | Description | PL-Link power consumption | Data points | Hardware catalog | As one instance in application library | Fully integrated in application library | In application types | Fully integrated in application library | Application types |
| Gamma | Sensor | UP 258D12 | Presence detector & brightness sensor, successor of UP258D11 | N/A | N/A | V4.0 | V4.0 | V4.0 | V4.0 | V4.0 | V4.0 |
| Gamma | Sensor | UP 258D31 | Presence detector, brightness, T sensor WIDE | 13 mA | 3 | V4.0 | V4.0 | No | No | No | No |
| Gamma | Sensor | UP 258D41 | Presence detector, brightness, T, rH. sensor WIDE pro | 13 mA | 4 | V4.0 | V4.0 | No | No | No | No |
| Gamma | Sensor | UP 258D51 | Presence detector, brightness, T, rH., CO2 sensor WIDE multi | 30 mA | 5 | V4.0 | V4.0 | No | No | No | No |
| Gamma | Sensor | UP 258D61 | Presence detector, brightness, T sensor WIDE DualTech | 20 mA | 3 | V4.0 | V4.0 | No | No | No | No |
| Gamma | Switch actuator | N 535D31 | Switching actuator with current detection: 4 channels | N/A | N/A | No | No | No | No | No | No |
| Gamma | Switch actuator | N 535D51 | Switching actuator with current | N/A | 8 | No | No | No | No | No | No |

| | | | | | | Supported workflows | | | | | |
|-------------------|-----------------|----------------------------|--|---------------------------|-------------|--|--|---|----------------------------|---|----------------------------|
| | | | | | | Room automation (without Critical Environment) | | | Critical Environment | | |
| | | | | | | ABT Pro version or higher | | | ABT Site version or higher | ABT Pro version or higher | ABT Site version or higher |
| Product portfolio | Product group | ASN | Description | PL-Link power consumption | Data points | Hardware catalog | As one instance in application library | Fully integrated in application library | In application types | Fully integrated in application library | Application types |
| | | | detection: 8 channels | | | | | | | | |
| Gamma | Switch actuator | N 535D61 | Switching actuator with current detection: 12 channels | N/A | 8 | No | No | No | No | No | No |
| Gamma | Switch actuator | N 53xD | Switching actuator 9 types: 4 / 8 / 12 channels | N/A | 4, 8, 12 | V3.2 | V4.0 | V4.0 | V4.0 | No | No |
| OpenAir | Damper actuator | GDB111.1E/KN | Damper actuator 5 Nm, non-spring return, AC 24 V, 150s | 5 mA | 2 | V4.0 | V4.1 | No | No | No | No |
| OpenAir | Damper actuator | GDB181.E/KN GLB181.E/KN | VAV compact controller | 5 mA | 2 | V4.0 | V4.0 | V4.0 | V4.0 | No | No |

| | | | | | | Supported workflows | | | | | |
|-------------------|-----------------|--------------|---|---------------------------|-------------|--|--|---|----------------------------|---|----------------------------|
| | | | | | | Room automation (without Critical Environment) | | | Critical Environment | | |
| | | | | | | ABT Pro version or higher | | | ABT Site version or higher | ABT Pro version or higher | ABT Site version or higher |
| Product portfolio | Product group | ASN | Description | PL-Link power consumption | Data points | Hardware catalog | As one instance in application library | Fully integrated in application library | In application types | Fully integrated in application library | Application types |
| OpenAir | Damper actuator | GLB111.1E/KN | Damper actuator 10 Nm, non-spring return, AC 24 V, 150s | 5 mA | 2 | V4.0 | V4.1 | No | No | No | No |
| Symaro | Sensor | AQR2532NNW | Flush-mounted room sensor with front module and base module | 5 mA | 1-3 | V4.0 | V4.0 | V4.0 | V4.0 | No | No |
| Symaro | Sensor | AQR2535NNW | Flush-mounted room sensor with front module and base module | 5 mA | 1-3 | V4.0 | V4.0 | V4.0 | V4.0 | No | No |
| Symaro | Sensor | AQR2535NNWQ | Flush-mounted room sensor with front module and base module | 5 mA | 1-3 | V4.0 | V4.0 | V4.0 | V4.0 | No | No |

23.3 Desigo Control Point

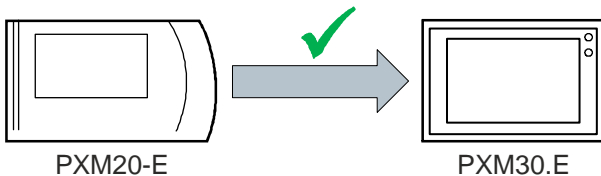
23.3.1 Compatibility with earlier systems

| System compatibility | PXM50-1 PXM40-1 PXM30-1 | PXM50.E PXM40.E PXM30.E | PXG3.W100-2 | PXG3.W200-2 |
|-----------------------------|-------------------------------|--|-------------|-------------|
| Desigo PX primary plants | n.a. | as of Desigo V4.0 | | |
| Desigo room automation PXC3 | n.a. | as of Desigo V5.0 | | |
| Desigo room automation DXR2 | n.a. | as of Desigo V6.0 | | |
| BACnet third-party devices | n.a. | as of BACnet revision 1.05 (≅ Desigo V4.0) | | |

23.3.2 Compatibility with earlier devices

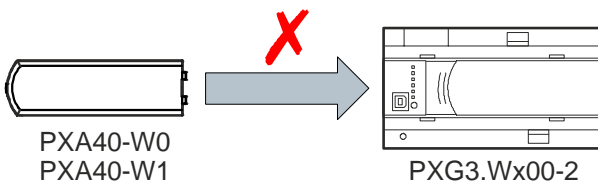
| Desigo devices ≤ Desigo V6.0 | Desigo Control Point | | | |
|------------------------------|-------------------------------|-------------------------------|-------------|-------------|
| | PXM50-1 PXM40-1 PXM30-1 | PXM50.E PXM40.E PXM30.E | PXG3.W100-2 | PXG3.W200-2 |
| PXG3.W100 | | | n.a. | n.a. |
| PXM20-E | | PXM30.E | | |
| PXM40 | PXM40-1 | | | |
| PXM50 | PXM50-1 | | | |

PXM20-E



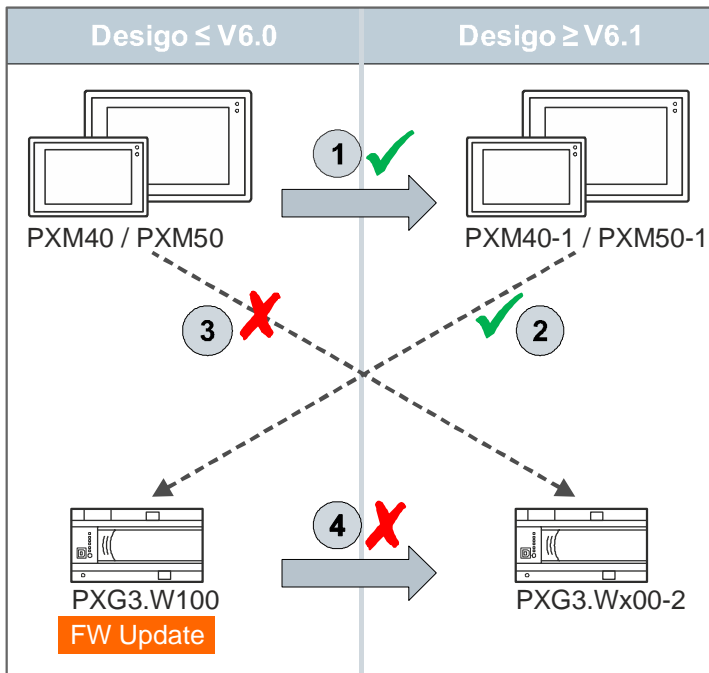
- The dimensions for the cut out are the same as for mounting in the panel.
- Supply voltage AC/DC 24 V.
- Ethernet connection for communication.
- No Power over Ethernet (PoE) connection on PXM30.E.

PX web



- PX web graphics are not compatible with graphics for the new web interfaces PXG3.Wx00-1 and -2.
- No workflow is currently available to automate migration of PX web graphics.
- New graphics can be efficiently created, based on templates or existing graphics.

Designo Touch and Web



- ① Same dimensions for the cut out as for mounting in the panel
Supply voltage AC/DC 24 V
Ethernet connection
Similar look and feel
- ② PXM40-1 and PXM50-1 panels are backward compatible with PXG3.W100 (PXG3.W100 FW updated required).
- ③ Existing PXM40 and PXM50 panels are not compatible with the PXG3.Wx00-1 and PXG3.Wx00-2 devices.
- ④ Engineering data, including graphics are not compatible with the new web interface.

23.3.3 Supported browsers

The following browsers support graphics and operation:

| | |
|---|--|
| Graphics editor | Google Chrome* Graphics can be created and edited without a tool using this browser. |
| Grade A Recommended web browser for standard operator units. | Google Chrome on desktop* Google Chrome on Android tablet* Google Chrome on Surface tablet* Microsoft Edge on Surface tablet <ul style="list-style-type: none"> Fully tested and supported browser. All functions are available and can be executed as documented. |
| Grade B Compatible web browser. | Google Chrome on Android cellphone* Microsoft Edge on desktop <ul style="list-style-type: none"> Fully tested and supported browser. Basic functions are available and can be executed as documented. Minor deviations in terms of display and operation to recommended browsers are possible (fonts, etc.). |
| Grade C Partially compatible standard web browsers. | Firefox Internet Explorer 11 All web browsers on iOS devices <ul style="list-style-type: none"> Minimally tested browsers. Not supported with Desigo Control Point. Access to the web server is possible in principle. |

* Chrome remains automatically on https when visiting an HTTPS page. It must be manually switched back to http to work with Desigo Control Point.

23.4 Upgrading to Desigo V6.3

We recommend that you upgrade to Desigo V6.3, to benefit from the newest features, improvements in hardware and software compatibility, the latest bug fixes, and to generally experience the best quality of our product.

23.4.1 Upgrading for Tool and Localization Managers in the Regional Companies (RC)

Upgrading from Desigo V6.2 Update 4 to V6.3 in the Regional Company (RC)

For details, see *Overview of Documentation for Experts* (CM110633en00_12) and *Project Converter Wizard* (CM110633en28_06).

1. Install the XWP V6.3 tool and library. Install the ABT Site/Pro V4.3 tool and libraries.
For details, see the release notes for the applicable tool: XWP, ABT Pro, and ABT Site.
2. Convert the RC local library (XWP and ABT Pro) and name it RC local library V6.3.
3. Install the RC local library V6.3 (XWP and ABT Pro).
For details, see *Library Maintenance Utility* (CM110625en_04).

23.4.2 Upgrading for Project Engineers

The following chapters describe the two typical use cases why and how a project engineer upgrades his Desigo projects.

23.4.2.1 Project case 1: Maintenance

Cases where you want to maintain your data and your site/project:

- Edit an existing site/project. Edit an existing site/project without adding automation stations or new features.
- Extend the site with new devices of the same type, while keeping the same old application version, and not using new features.
- Replace a defect automation station with a new device of the same kind, where the old application can be reused, e.g., replace a PX with a PX, a PXC3 with a PXC3, a DXR2 with a DXR2.
The downgrade of automation station firmware is a potential security risk. Doing so is strongly discouraged.
- Benefit from the latest security releases. Upgrade the firmware of the devices. Keep the applications as they are.

Prerequisites for upgrading from Desigo V6.2 Update 4 to V6.3 (Offline)

- The XWP V6.3 Tool and library and the ABT V4.3 Site/Pro Tool and libraries are installed.
For details, see the release notes for the applicable tool: XWP, ABT Pro, and ABT Site.
- The needed RC local library V6.3 is installed.
For details, contact your local Tool Manager.
- You have the project data and the credentials for the project you want to work on.
- The project data conversion from V6.2 Update 4 to V6.3 happens automatically.
 - When you open an XWP project, the XWP project is converted automatically.
 - When you open an automation station in XWP, the CFC data is converted automatically.
 - When you navigate to ABT Site, the conversion is done automatically.

Once XWP data and/or ABT data have been converted, you may not open the project with an old tool version. The new TIA15.1 version is included with the ABT V4.3 Tools installation. Conversion is needed. For details, see *Project Converter Wizard* (CM110633en28_06).

How to upgrade from Desigo V6.2 Update 4 to V6.3 (Online)

1. Upgrade the automation stations firmware to the latest version or replace the automation station with a new automation station.
 - Read back the parameters of the device with the tool, when the device is available for readback.
 - Check the availability of the latest firmware per device in the VVS table.For details about upgrading the automation station firmware, see the documentation of the applicable tool: XWP, ABT Pro, and ABT Site.
2. Edit the application in the automation station.
For details, see the documentation of the applicable tool: XWP, ABT Pro, and ABT Site.
3. You do not need to upgrade Desigo CC and/or Desigo Control Point.

23.4.2.2 Project case 2: Extension with new features

Cases where you want to use new features for one or several automation stations:

- Extend the site with new devices from the latest version (recently introduced).
- Benefit from new features both on firmware and applications (recently introduced).
- Replace an automation station type with another automation station type, which does not have the possibility to fully reuse the application program, and needs to be reprogrammed.

Prerequisites for upgrading from Desigo V6.2 Update 4 to V6.3 (Offline)

- The XWP V6.3 Tool and library and the ABT V4.3 Site/Pro Tool and libraries are installed.
For details, see the release notes for the applicable tool: XWP, ABT Pro, and ABT Site.
- The needed RC local library V6.3 is installed.
For details, contact your local Tool Manager.
- You have the project data and the credentials for the project you want to work on.
- The project data conversion from V6.2 Update 4 to V6.3 happens automatically.
 - When you open an XWP project, the XWP project is converted automatically.
 - When you open an automation station in XWP, the CFC data is converted automatically.
 - When you navigate to ABT Site, the conversion is done automatically.

Once XWP data and/or ABT data have been converted, you may not open the project with an old tool version. The new TIA15.1 version is included with the ABT V4.3 Tools installation. Conversion is needed. For details, see *Project Converter Wizard* (CM110633en28_06).

How to upgrade from Desigo V6.2 Update 4 to V6.3 (Online)

1. Upgrade the automation station firmware and the application to the latest version.
 - Read back the parameters of the device with the tool.
 - Check the availability of the latest firmware per device in the VVS table.
For details about upgrading the automation station firmware, see the documentation of the applicable tool: XWP, ABT Pro, and ABT Site.
 - Upgrade the application type (ABT Site).
 - Upgrade the device type (ABT Pro).
For details about upgrading the application program, see the documentation of the applicable tool: XWP, ABT Pro, and ABT Site.
 - Make specific changes according to the new features.
 - Run a full compile and load the upgraded program.
2. If you upgrade from a lower version than V6.2, run the global commanding script for the alarming improvements.
3. Upgrade Desigo CC and/or Desigo Control Point.
To support new features in the system, the Desigo CC and/or Desigo Control Point version must correspond to the highest installed version. e.g., updates of functions included in Desigo V6.3 need the latest Desigo Control Point release version from Desigo V6.3 to function fully.

23.4.3 Upgrading restrictions

Branch Office Server (BOS)

Desigo V6.3 is compatible with Branch Office Server (BOS) V6.2 Update 4, but we recommend that you upgrade to BOS V6.3.

Desigo PXR / LonWorks system controller

A migration of previously programmed and operational V2.2 - V2.37 system controllers PXR11/12 to Desigo V6.3 using PXC00(-E).D+PXX-L11/12 is required:

- To use LNS based LonWorks standard tools NL220 (Newron System) or LonMaker (Echelon) as an alternative to RXT10.5. This applies to projects based on LNS TE and OpenLNS.
- When the runtime system (project) requires the use of certified devices with BACnet rev. 1.12.

There is no need to exchange existing PXR11/12 devices. Migration to PXC00(-E).D with a PXX-L... is only required if the aforementioned conditions are required.

For more information, see chapter *Upgrade PXR11/12* in *Desigo Xworks Plus Online Help* (CM111006).

23.5 Siemens WEOF clients

This information is only for Siemens employees who use a WEOF client computer.

Desigo software

All Desigo software programs and LibSets (LED) operate on the Siemens WEOF client.

| Minimum user level required | Desigo product |
|-----------------------------|--|
| Standard User | Desigo Configuration Module (DCM) |
| Permanent Open User | Desigo Xworks Plus (XWP) including PX firmware library (FW), Automation Building Tool (ABT) and additional tools |
| Permanent Open User | Branch Office Server (BOS) |
| Permanent Open User | RXT10 (including RX library) |
| Permanent Open User | Headquarter (HQ) and Regional Company (RC) libraries |

Third-party engineering software

For information about ETS, see:

<https://www.knx.org/knx-en/for-professionals/software/ets-5-professional/index.php>

For information about IzoT CT, see:

<https://www.echelon.com/resource-library-results?filters=&q=lonmaker>

23.6 Migration compatibility

Migration of Xworks Plus (XWP)

| Described in | Requirements |
|--------------|--|
| CM110776 | Automation Level Engineering Manual |
| CM110563 | Replacement of legacy I/O modules by TX-I/O modules or workarounds |

Migration of Unigr

| Described in | Requirements |
|--------------|---|
| CM110496 | Unigr tools V7.61 with Unigr automation level V7.64 |

Migration of Integral

| Described in | Requirements |
|--------------|--|
| CM110499 | NCRS from V3.1 (only automation level) |
| CM110498 | NITEL from V1.31 (only automation level) |

For replacing Integral RS modules (NRUA, NRUB, NRUC, and NRUD) with PXC automation stations and PXC-NRUD modules, Desigo supports the use of PXC-NRUD modules with PXC100/200(-E).D and PXC50(-E).D.

Migration of Visonik

| Described in | Requirements |
|--------------|--|
| CM110497 | DCS from V22.16 Patch 195 or V24.16 Patch 195 (server with automation level) |

23.7 Hardware requirements of Desigo software products

| Product | CPU | Frequency | Storage | Hard disk | Other |
|--|--|------------------------------------|---------------------------------------|--|---|
| ABT Site | Compatible with Intel and AMD technology | > 2.0 GHz (> 3 GHz recommended) | 16 GB RAM | > 100 GB SSD or HDD with very good performance. The greater the number and size of the projects, the more additional memory is required. An ABT project size may be anywhere between 250 MB and 30 GB. | Monitor: 1680x1050 recommended |
| Desigo Xworks Plus (and all other utilities) incl. ABT Site/Pro on the same computer | Compatible with Intel and AMD technology | > 2.0 GHz (> 3 GHz recommended) | 8 GB RAM (> 16 GB RAM recommended) | 50 GB HD* with good performance (HDD at very fast access times) | Monitor: 1366x768 DVD (SSD drive) (USB port for SSA to Discovery Network Tool (DNT) as alternative to ethernet connection) Multiple core processors, e.g., for VMware |
| Desigo Configuration Module (DCM) | Compatible with Intel and AMD technology | 1.6 GHz | 1 GB RAM | 40 GB HD | |
| Branch Office Server (BOS) | Compatible with Intel and AMD technology | > 1.6 GHz (2.5 GHz) | 8 GB RAM recommended | HD size depending on project data volume | |
| RXT10.5 | Compatible with Intel and AMD technology | > 1.6 GHz | 4 GB RAM | HD size depending on project data volume | |

Key

* Desigo Xworks Plus (XWP) requires ca. 1.4 GB memory. Automation Building Tool (ABT) requires ca. 1-2 GB memory. Uncompressed project data requires an additional 0.5 MB memory per data point (reference value). The performance depends on available memory.

The indicated values apply to a host installation. For stable and reliable operation of VMware, CPU and RAM requirements are higher.

The recommended values allow for larger projects (up to 12 PXC3 with 8 rooms each per ABT project). For details, see chapter *Compatibility with Operating Systems*.

Configure SSDs for a long life. See Microsoft documentation.

ABT projects require ca. 2.5 times more memory per PXC3 room automation station compared to PXC automation stations.

Parallel port or USB port for license dongle.

For online functions you need:

- LonWorks interface card or LonWorks dongle
- Ethernet interface
- Connection cable for automation stations
- USB port for P-bus BIM and SSA Discovery Network Tool (DNT) connection

The following software is required:

- Operating system: See chapter *Compatibility with Operating Systems*
- Microsoft Office: See chapter *Compatibility with Microsoft Office*
- Acrobat Reader 6.0 or higher (optional installation with tool installation)
- WinZIP
- .NET Framework >= V3.5 (version 3.5 is available on the tool DVD)

24 Desigo PXC4 and PXC5

PXC4 & PXC5 Range Overview (A6V11973782)

Description of the range for a small system with:

- Desigo Control Point embedded management station
- Desigo Control Point touch panel range
- PXC4 automation station with I/O extension modules
- PXC5 system controller

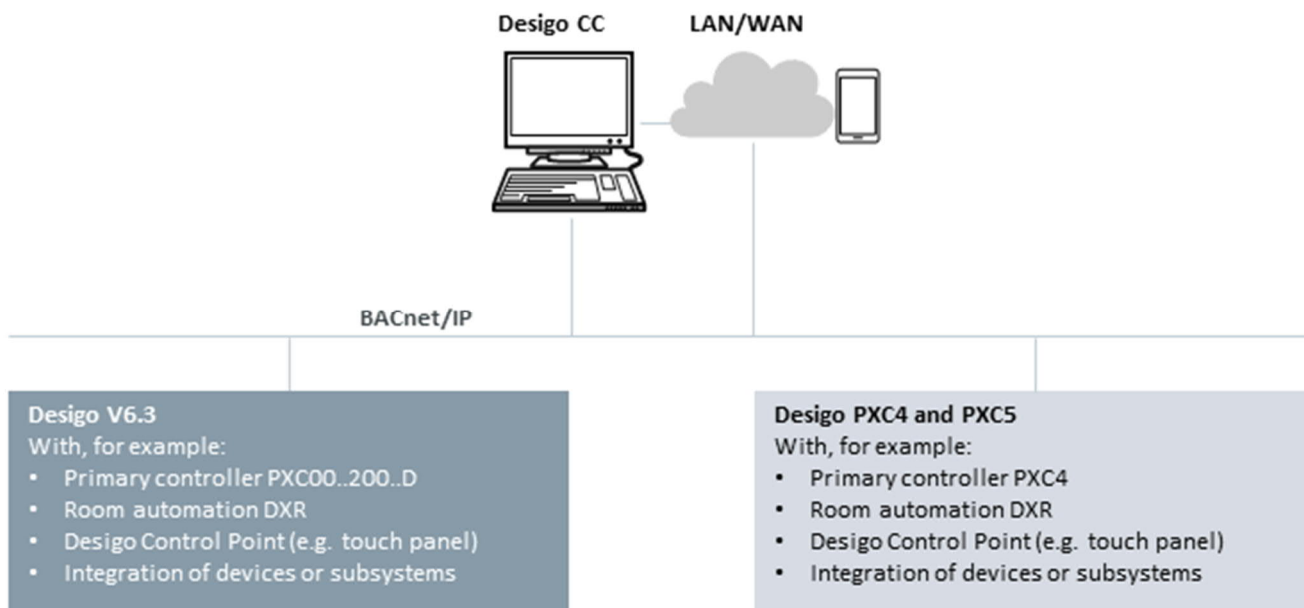
PXC4 & PXC5 Planning Overview (A6V11973797)

Includes the following topics:

- Planning guidelines
- Overview of compatible products
- Various typical topologies
- Technical limitations

25 Compatibility of Desigo V6.3 with PXC4 and PXC5

Overview



Online system Desigo V6.3 with PXC4 and PXC5

- Full BACnet communication between the Desigo CC management platform and the Desigo V6.3 system.
- Full BACnet communication between the Desigo CC management platform and the Desigo PXC4 and PXC5 system.
- BACnet communication between Desigo V6.3 and Desigo PXC4 and PXC5: Via COV's.

For system limits of Desigo V6.3, see Compatibility [→ 320].

For system limits of Desigo PXC4 and PXC5, see *PXC4 & PXC5 Planning Overview (A6V11973797)*.

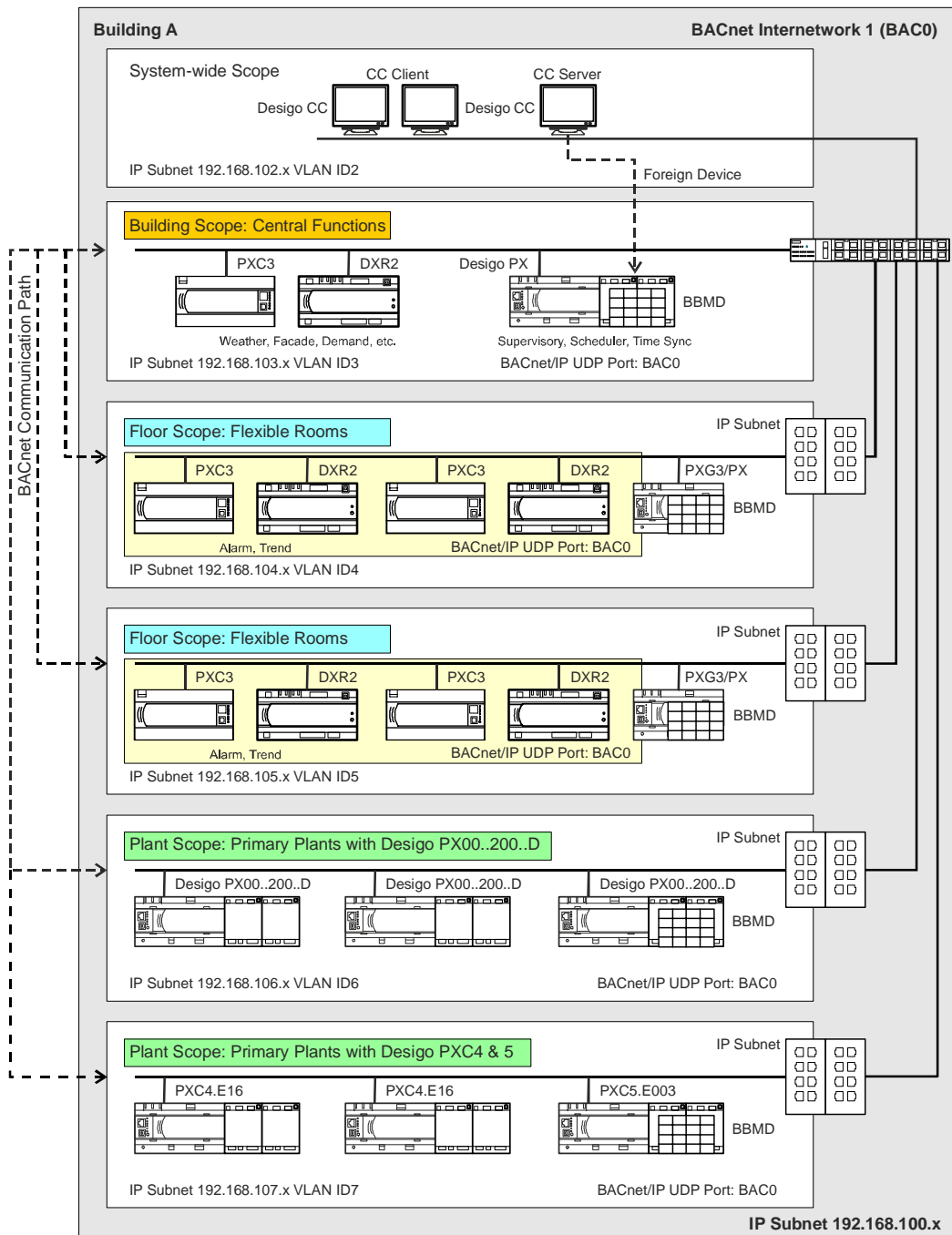
Prerequisites for extending Desigo V6.3 (XWP project) with PXC4 and PXC5

- Upgrade the management platform Desigo CC to V5.0.
- Make sure all PXC00...200..D primary controllers and room automation products adhere to the Desigo V6.3 compatibility.
- You are using Desigo Xworks Plus (XWP) V6.3.
- You are using ABT Site V4.3.

Engineering of Desigo V6.3 projects with PXC4 and PXC5

Network topology

Structure the network topology as follows:



For more information, see *Application Guide for BACnet Networks in Building Automation (A6V11159798)*.

Engineering and commissioning of Desigo PXC4 and PXC5

Engineer and commission the Desigo PXC4 and PXC5 subsystem with ABT site V4.3 and export/import it into the Desigo CC management platform.

For more information on the system compatibility of the Desigo CC management platform, see *Desigo CC System Description (A6V10415500)*.

Engineering and commissioning of Desigo V6.3

Engineer enhancements and adaptations in Desigo V6.3 with XWP V6.3 (ABT Site) and export/import it into the Desigo CC management platform.

For more information on the system compatibility of the Desigo CC management platform, see *Desigo CC System Description (A6V10415500)*.

Data exchange between Desigo V6.3 and Desigo PXC4 and PXC5 on automation level

Similar to the integration of third-party systems via BACnet references into the Desigo V6.3 system engineered via XWP V6.3.

For more information, see *XWP Online Help*.

Issued by
Siemens Switzerland Ltd
Smart Infrastructure
Global Headquarters
Theilerstrasse 1a
CH-6300 Zug
+41 58 724 2424
www.siemens.com/buildingtechnologies