wieland

samos® PRO

samos® PRO COMPACT-Gateways

Manual

Doc. no. BA000970

Last Update: 07/2016 (Rev. E)

Info

Copyright

Subject to change.

Subject to technical changes for reasons of continued development.

Table of Contents

1 About this document

Please read this section carefully before you work with these operating instructions and the samosPRO COMPACT gateway system.

1.1 Function of this document

This manual is only valid in combination with the other samosPRO manuals (see *Overview of manuals and operating instructions [ch. 1.2, p. 8]*). It instructs the technical staff of the machine manufacturer or machine operator in the safe installation, configuration, electrical installation, commissioning, operation and maintenance of the samosPRO gateway.

This manual does **not** provide operating instructions for the machine, which incorporates modular samosPRO safety controls and a samosPRO gateway. Information in this regard is provided in the operating instructions for each machine.

1.2 Overview of manuals and operating instructions

The following documents have been provided for the samosPRO COMPACT system, with clearly distinguishable applications and installation instructions for each of the modules.

- This manual (BA000970) describes all samosPRO COMPACT gateways and their functions in detail.
- The "samosPRO hardware" manual (BA000966) describes all samosPRO modules that can be used in combination with COMPACT modules and their functions in detail. The hardware manual should especially be used when configuring the samosPRO safety controls.
- The "samosPLAN5+ software" manual (BA000968) describes the software-supported configuration and parameterization of the samosPRO safety controls with COMPACT modules. In addition, the software manual contains a description of the important diagnostic functions for operation and detailed information for identifying and eliminating errors. Use the software manual mainly when configuring, commissioning and operating samosPRO safety controls.
- Each samosPRO module contains the installation instructions/brief instructions. These instructions provide information on the fundamental technical specifications of the modules and contain simple installation instructions. Use the installation instructions/brief instructions when installing the samosPRO safety control.

This manual contains original operating instructions in accordance with the Machinery Directive.

1.3 Target audience

This manual is aimed at the planners, developers and operators of systems that incorporate modular samosPRO COMPACT safety controls and that want to exchange data with a field bus (controls) via a gateway.

It is also aimed at persons commissioning a samosPRO COMPACT gateway system for the first time or maintaining such a system.

1.4 Information depth

This manual contains information about the following topics related to samosPRO COMPACT gateways:

- Installation
- Integration into the network
- Configuration using the samosPLAN5+ configuration software
- Data transmission to and from the network
- State information, projection and associated mapping
- Item numbers

Important information

ATTENTION

Safety information and protective measures
The safety instructions and precautions for use of samosPRO COMPACT gateways must be adhered to!

NOTICE

Also consult our website on the Internet at the following link: (http://eshop.wieland-electric.com).
There you will find the following files available for download:
- GSD file of the SP-PROFIBUS-DP for PROFIBUS-DP
- EDS file for SP-CANopen for CANopen

1.5 Scope of application

This manual is valid for the samosPRO COMPACT gateway modules SP-COP2-ENI, SP-PROFIBUS-DP and SP-CANopen.

1.6 Glossary

Term	Explanation
Data block	A data block contains 2-12 bytes of the relevant data set (depending on the gateway used).
Data set	Describes a quantity of associated data, e.g. logic values or system state data. A data set can consist of several data blocks.
SINT	Short integer = 1 byte
PLC	Programmable Logic Controller (PLC)
UDINT	Unsigned double integer = 4 bytes = 2 words
UINT	Unsigned integer = 2 bytes = 1 word

1.7 Symbols/icons and writing style/spelling standard used

NOTICE

These are notes that provide you with information regarding particularities of a device or a software function.

ATTENTION

Warning!
A warning lets you know about specific or potential hazards. It is intended to protect you from accidents and help prevent damage to devices and systems.

- **Please read and follow the warnings carefully!**
  Failure to do so may negatively impact the safety functions and cause a hazardous state to occur.

Menus and commands

The names of software menus, submenus, options, and commands, selection fields, and windows are written in **bold font**. Example: Click on **Edit** in the **File** menu.

2 Safety

This section is intended to support your safety and the safety of the system users.

➡ Please read this section carefully before you work with the modular samosPRO safety control or with a machine protected by a samosPRO.

2.1 Qualified persons

The samosPRO COMPACT gateway may only be installed, commissioned, and maintained by qualified persons.

Qualified persons are those who

- have suitable technical training and
- have been trained by the machine operator in the operation and applicable safety guidelines and
- have access to the manual for the samosPRO COMPACT gateway and the samosPRO COMPACT modular controls and have read and taken note of them.

2.2 Proper use

The samosPRO COMPACT gateways can only be operated in combination with a samosPRO COMPACT system. The firmware version of the linked SP-COP modules must be at least V1.0.0, while the samosPLAN5+ configuration software version must be at least 1.0.0.

The samosPRO COMPACT gateways do not have their own power supply.

ATTENTION

The samosPRO COMPACT gateways are not suitable for operation with a safety field bus!
These gateways do not only generate safety-related field bus data (state bytes) for control and diagnostic purposes.

ATTENTION

Do not use data from a samosPRO COMPACT gateway for safety-related applications!
The samosPRO COMPACT gateway can be used to integrate non-safety-related data into the logic editor in such a way that the safety function of the samosPRO COMPACT system may be adversely affected. Never integrate a gateway into a samosPRO COMPACT system without having this source or risk checked by a safety specialist.

These modules may only be operated by qualified staff and may only be used on a machine on which they have been installed and commissioned for the first time by a qualified person in accordance with this manual.

ATTENTION

The safety instructions and precautions for use of samosPRO COMPACT gateways must be adhered to!
In the event of any other use or any changes to the device – including within the scope of installation – this shall nullify any sort of warranty claim with respect to Wieland Electric GmbH.

NOTICE

- Please follow the standards and guidelines valid in your country when installing and using samosPRO COMPACT gateways.
- The national/international legal regulations apply to the installation and use of the modular samosPRO COMPACT safety controls as well as for the commissioning and repeated technical testing, particularly the following:
  – EMC Directive 2004/108/EC,
  – the Use of Work Equipment Directive 2009/104/EC,
  – the accident prevention / safety regulations.
- The manual must be provided to the operator of the machine on which the samosPRO COMPACT system is to be used. The machine operator must be trained by qualified persons and is required to read this manual.

ATTENTION

May only be used in an industrial environment
The samosPRO COMPACT system fulfills the requirements for class A (industrial applications) according to the "Emitted interference" basic trade standard. The samosPRO COMPACT system is therefore only suitable for use in an industrial environment.

2.3 Environmentally friendly behavior

samosPRO COMPACT gateways have been designed in such a way that they pollute the environment as little as possible and only consume a minimum of energy and resources.

➡ Make sure that you also carry out work while always considering the environment.

2.3.1 Disposal

The disposal of unusable or irreparable devices should always be done in accordance with the respectively valid country-specific waste-elimination guidelines (e.g. European Waste Code 16 02 14).

NOTICE We will be happy to help you in disposing of these devices. Simply contact us.

2.3.2 Sorting of materials

ATTENTION

The sorting of materials may only be carried out by qualified persons!
Care must be used when disassembling the devices. There is a risk of injuries.

Before you can route the devices to the environmentally-friendly recycling process, it is necessary to sort the various samosPRO materials.

➡ Separate the housing from the rest of the components (particularly from the PC board).

➡ Place the separated components into the corresponding recycling containers (see the following table).

Table 1: Overview of disposal according to components

Components	Disposal
Product Housings, PC boards, cables, connectors, and electric connecting pieces	Electronics recycling
Packaging Cardboard, paper	Paper/cardboard recycling

3 Product description for samosPRO COMPACT gateways

samosPRO COMPACT gateways allow the samosPRO COMPACT system to transmit non-safety-related data for control and diagnostic purposes to the external field bus system and to receive them.

NOTICE

Where not otherwise indicated, this manual always considers the data exchanged between the samosPRO COMPACT system and the relevant network from the point of view of the network master (PLC). Thus data sent to the network from the samosPRO COMPACT system are termed input data, while data received from the network are termed output data.

ATTENTION

Do not operate a samosPRO COMPACT gateway on a safety field bus!
The samosPRO COMPACT gateway modules are not suitable for operation with a safety field bus! They do not support any safety mechanisms that would be required for communication within a safety network.

Configuration of samosPRO COMPACT gateways takes place via the samosPLAN5+ configuration software, using a PC or Notebook connected to the SP-COPx main module via the USB interface or RJ45 Ethernet interface.

The safety-related logic of the samosPRO COMPACT system works independently of the gateway. However, if the samosPRO COMPACT system has been configured in such a way that non-safety-related information from the field bus can be integrated into the logic editor, switching off the gateway may result in availability problems.

A samosPRO COMPACT gateway can only be operated on a samosPRO COMPACT system. It does not have its own power supply. A maximum of two samosPRO COMPACT gateways can be operated simultaneously for each system.

The gateway for Modbus TCP, PROFINET IO or EtherNet/IP is integrated into the SP-COP2-ENI main module, while the gateways for Profibus-DP, CANopen or EtherCAT are housed in a 22.5 mm wide installation housing for 35 mm standard rails in accordance with EN 60715.

Order information: *Order data [ch. 13, p. 181]*

3.1 Version, compatibility, and features

There are various module versions and function packages for the samosPRO product family that enable various functions. This section will give you an overview as to which module version, which function package, and/or which version of the samosPLAN5+ you will need to be able to use a certain function or a certain device.

Table 2: Module and software versions required

	Available from module version			
Feature / functionality	**SP-COP1-x**	**SP-COP2-EN-x**	**SP-COP2-ENI-x**	**samosPLAN5+**
Safe I/O (SP-SDIO, SP-SDI)	A-01	A-01	A-01	V1.0
Modbus TCP	---	---	A-01	V1.0
Profinet IO	---	---	B-01.xx	V1.2
Non-secure I/O (SP-DIO)	C-01.xx	C-01.xx	C-01.xx	V1.3
EtherCAT (SP-EN-ETC)	C-01.xx	C-01.xx	C-01.xx	V1.3
EtherNet/IP	---	---	D-01.xx	V1.4
Press functions[1)]	D-01.xx	D-01.xx	D-01.xx	V1.4
[1)] only available with module variants **-P** (example: SP-COP2-EN-**P**-x)				

Info

- You will find the module version on the type plate of the samosPRO modules.
- You can find the samosPLAN5+ version in the green **File menu** under **About**.
- You can obtain the latest version of the samosPLAN5+ on the Internet at http://www.wieland-electric.de.
- Newer modules are backwards-compatible, which means that each module can be replaced with a module having a higher module version.
- You can find the date of manufacture for a device on the type plate in the S/N field in the format <Product no.>yywwnnnnn (yy = year, ww = calendar week).

3.2 Equipment variants

There are three samosPRO COMPACT gateways for various network types. The Modbus TCP / PROFINET IO and EtherNet/IP gateway of the SP-COP2-ENI main module or the external SP-EN-ETC gateway are suitable for Ethernet networks. The SP-PROFIBUS-DP gateway and the SP CANopen gateway are external field bus gateways without an Ethernet function.

Table 3: Equipment variants and their main characteristics

Gateway	Network type	Ethernet TCP/IP socket interface
SP-COP2-ENI-x	Modbus TCP with master and slave operation	Client/Server on Port 502
SP-COP2-ENI-x	PROFINET IO device	
SP-COP2-ENI-x	EtherNet/IP device	
SP-PROFIBUS-DP	PROFIBUS DP slave	–
SP-CANopen	CANopen slave	–
SP-EN-ETC	EtherCAT slave	

NOTICE You will find the manufacturing date of a device on the type label in the S/N field in the format yywwnnnn (yy = year, ww = calendar week, nnnn = consecutive serial number within a calendar week).

3.3 Data transferred to the network (network input data sets)

Available data

The samosPRO COMPACT gateways can provide the following data:

- Operating data
  - **Logic results of** samosPRO COMPACT (see *Routing table [ch. 5.1.2, p. 33]*)
  - **Input values** (HIGH/LOW) for all samosPRO COMPACT input expansion modules in the system
  - **Output values** (HIGH/LOW) for all samosPRO COMPACT input/output expansion modules (see *Module state / input and output values [ch. 3.3.1, p. 17]*)
  - **Output data** from another network, i.e. data received from a second gateway in the samosPRO COMPACT system (see *Transmission of data from a second network [ch. 3.3.3, p. 18]*)
- Diagnostics
  - **Test values** (CRCs) (see *Configuration test values (CRCs) [ch. 3.3.4, p. 18]*)
  - **Error and state information** for all modules except SA-OR-S2 and SA-OR-S1 (see *Error and state information for modules [ch. 3.3.5, p. 19]*)

Data sets

The physical samosPRO COMPACT modules are not presented as typical hardware modules in the network. Instead, the data provided by the samosPRO COMPACT system have been arranged in four *input data sets*.

- **Data set 1** (max. 50 bytes) contains the operating data. It can be compiled with the aid of samosPLAN5+. In the form in which it is delivered, the content of data set 1 is preconfigured; it can be freely modified.
  Details: see table *"Overview of input data sets" [ch. 3.3, p. 15]*

  For the SP-PROFIBUS-DP, data set 1 was divided into five input data blocks, with data blocks 1–4 each containing 12 bytes and data block 5 two bytes.
  For the SP-CANopen, data set 1 was divided into four blocks, each with 8 bytes.
  You will find more detailed information in the corresponding section for each gateway.
- **Data set 2** (32 bytes) contains the test values (CRCs) for the system configuration.

  See table "Overview of input data sets 1-3 (basic settings for Modbus TCP)" below
- **Data set 3** (60 bytes) contains the state and diagnostic data for the various modules, with four (4) bytes per module, with the main module comprising 3 x 4 bytes. Details: see table *"Meaning of module state bits" [ch. 3.3.5, p. 19]*
- **Data set 4** (60 bytes) is currently filled with reserved values.

The following table provides an overview of which data sets are provided by which gateway.

Table 4: Availability of data sets 1–4

	Data set 1	Data set 2	Data set 3	Data set 4
SP-COP2-ENI	Modbus TCP PROFINET IO EtherNet/IP	Modbus TCP PROFINET IO EtherNet/IP	Modbus TCP PROFINET IO EtherNet/IP	Modbus TCP PROFINET IO
SP-EN-ECT	EtherCAT	EtherCAT	EtherCAT	-
SP-PROFIBUS-DP	PROFIBUS DP	–	–	–
SP-CANopen	CANopen	CANopen (SDOs)[1)]	CANopen (SDOs)[1)]	–
[1)] The SP-CANopen is used to provide diagnostic data via CANopen SDO (service data objects). More information about how to provide state and diagnostic data with the aid of the CANopen gateway may be found here: *CANopen gateway [ch. 10, p. 118]*				
[2)] Readable with instance 2 of class 120				
[3)] Readable with instance 3 of class 120 and byte 52 to 111 of assembly 167				

Table 5: Overview of input data sets 1–3 (basic setting for Modbus TCP)

	Data set 1	Data set 2	Data set 3	Data set 4
Byte 0	Input values for Module 0 (I1..I8)	Project CRC	Module state SP-COPx	Reserved
Byte 1	Input values for Module 0 (I9..I16)		Module state SP-COPx	
Byte 2	Input values for Module 0 (IQ1..IQ4)		Test impulse comparison for SP-COP inputs	
Byte 3	Output values for Module 0 (Q1..Q4, IQ1..IQ4)		Test impulse comparison for SP-COP inputs	
Byte 4	Direct data (Off) 0	System CRC (PROFIBUS DP and EtherCAT)	Test impulse comparison for SP-COP inputs	
Byte 5	Direct data (Off) 1		State of two-channel SP-COP inputs	
Byte 6	Direct data (Off) 2		State of two-channel SP-COP inputs	
Byte 7	Direct data (Off) 3		Reserved	
Byte 8	Direct data (Off) 4	Reserved	Stuck-at error at SP-COP outputs	

	Data set 1	Data set 2	Data set 3	Data set 4
Byte 9	Direct data (Off) 5		Stuck-at error at SP-COP outputs	
Byte 10	Direct data (Off) 6		Reserved	
Byte 11	Direct data (Off) 7		Reserved	
Byte 12	Input values for Module 1		State of Module 1	
Byte 13	Input values for Module 2		State of Module 1	
Byte 14	Input values for Module 3		State of Module 1	
Byte 15	Input values for Module 4		State of Module 1	
Byte 16	Input values for Module 5		State of Module 2	
Byte 17	Input values for Module 6		State of Module 2	
Byte 18	Input values for Module 7		State of Module 2	
Byte 19	Input values for Module 8		State of Module 2	
Byte 20	Input values for Module 9		State of Module 3	
Byte 21	Input values for Module 10		State of Module 3	
Byte 22	Input values for Module 11		State of Module 3	
Byte 23	Input values for Module 12		State of Module 3	
Byte 24	Output values for Module 1		State of Module 4	
Byte 25	Output values for Module 2		State of Module 4	
Byte 26	Output values for Module 3		State of Module 4	
Byte 27	Output values for Module 4		State of Module 4	
Byte 28	Output values for Module 5		State of Module 5	
Byte 29	Output values for Module 6		State of Module 5	
Byte 30	Output values for Module 7		State of Module 5	
Byte 31	Output values for Module 8		State of Module 5	
Byte 32	Output values for Module 9	Not available	State of Module 6	
Byte 33	Output values for Module 10		State of Module 6	
Byte 34	Output values for Module 11		State of Module 6	
Byte 35	Output values for Module 12		State of Module 6	
Byte 36 … Byte 47	Not allocated		State of Module 7 … Status of Module 9	
Byte 48			State of Module 10	
Byte 49			State of Module 10	
Byte 50	Not available		State of Module 10	
Byte 51			State of Module 10	
Byte 52 … Byte 55			State of Module 11 … Status of Module 11	
Byte 56			State of Module 12	
Byte 57			State of Module 12	
Byte 58			State of Module 12	
Byte 59			State of Module 12	
Length	50 bytes	32 bytes	60 bytes	60 bytes

NOTICE When two-channel input or output elements have been configured for an I/O module, only the lowest bit constitutes the input or output state (on/off) for the corresponding element. It is represented by the tag name of the element. The highest bit represents the state of this input/output.

NOTICE The input values in data set 1 do not represent the physical state at the input terminals, but the pre-processed input values that are used for logic processing.

3.3.1 Direct gateway output values

It is possible to write values directly from the logic editor to a gateway. Four bytes have been reserved for this purpose in the basic settings for data set 1; however, up to the total number of 50 bytes of data set 1 may be configured as direct gateway output values. You can obtain additional information at: *Direct gateway output values [ch. 5.3, p. 40]*.

3.3.2 Module state / input and output values

The samosPRO COMPACT gateways can transmit the input and output states of all samosPRO COMPACT modules connected to the samosPRO COMPACT system to the network. Data set 3 contains a non-modifiable configuration. Moreover, data set 1 can be adapted to contain up to 4 bytes of collective state information. Only the input and output values for data set 1 have been predefined and these can be freely adapted. You will find more detailed information in the section on the relevant gateway, as well as in the following section: *Configuration of gateways with samosPLAN5+ [ch. 5, p. 31]*

Module state

The samosPRO COMPACT gateways can transfer the state of the linked modules to the network. A total of 4 bytes are available for this purpose.

Table 6: Module state

Module state	Size	Meaning	Assignment
Input data state	2 bytes	One sum bit per module for the state of the module inputs 0 = error 1 = no error	Bit 0 = SP-COPx Bit 1 = 1. Extension module Bit 2 = 2. Expansion module … Bit 13 = 1. Gateway Bit 14 = 2. Gateway Bit 15 = reserved
Output data state	2 bytes	One sum bit per module for the state of the module outputs 0 = error 1 = no error	

You will find information about the meaning of the state bits at: "samosPLAN5+ Software" manual (BA000968), "Status bits for type SP-COP modules (reference)"

NOTICE The input and output state of the SP-SDI and SP-SDIO modules is only available from firmware state V2.00.0.

Input and output values for the modules

- Input values for I/O modules
  1 byte for data set 1 is available for every expansion module. The input values show the state of the preliminary evaluation of the I/O module. This corresponds to the state of the element in the main module logic. The level at the associated terminal cannot be clearly detected from this, as the data may be set to low, irrespectively of the level at the input terminal, by means of the cross-connection detection or two-channel evaluation (e.g. I1-18).
  When two-channel input elements have been configured for an I/O module, only the lower-value bit represents the pre-evaluation state of the corresponding element (e.g. bit 0 for I1 and I2, bit 2 for I3 and I4, bit 4 for I5 and I6, bit 6 for I7 and I8). The higher-value bit (bit 1, 3, 5 and 7) is used as follows in this case:
  0 = error 1 = no error
- Output values for I/O modules
  1 byte for data set 1 is available for every module with outputs. The output values indicate the state of the control information from the logic of the main module for the relevant element of the I/O module. The level of the associated terminals cannot be clearly detected from this, as the output may be switched off via the cross-connection detection or the overload connection function.
  When two-channel output elements have been configured for an I/O module, only the lower-value bit represents the control information (e.g. bit 0 for Q1 and Q2, bit 2 for Q3 and Q4, bit 4 for Q5 and Q6, bit 6 for Q7 and Q8). The higher-value bit (bit 1, 3, 5 and 7) is not used as follows in this case (low):

3.3.3 Transmission of data from a second network

If your samosPRO COMPACT system contains two gateways, it is possible to forward information which the first gateway receives from a network (e.g. from a Modbus PLC) via the second gateway to a second network (e.g. to a PROFIBUS master) and vice versa.

3.3.4 Configuration test values (CRCs)

Data set 2 contains the following configuration test values for the samosPRO COMPACT system:

- Project-CRC of the project file created with samosPLAN5+
- System-CRC, uniquely assigned to a module version, consisting of internal software and hardware version

The CRCs are each 4 Bytes in length. Data set 2 can be read only.

The project CRC with Modbus/TCP is transmitted in Big Endian format.

System-CRC available starting with module version B-01.01 for PROFIBUS DP and EtherCAT.

3.3.5 Error and state information for the modules

Data set 3 and 4 contain the state information for the modules that will be transferred to the network.

Ten bytes are transferred for each SP-COPx main module. For each SP-SDI and SP-SDIO I/O module, four bytes are transmitted in the little endian format, e.g. as a 32-bit word, with the first byte being placed into the least significant byte of the whole number (extreme left) and the fourth byte into the most significant byte of the whole number (extreme right).

Data sets 3 and 4 cannot be adapted.

Module state bit of main module SP-COPx

The module state bits have the following meaning, if not otherwise indicated:

0 = error

1 = no error

Reserved bits have the value 1

Table 7: Meaning of module state bits of main module SP-COPx (only for Modbus)

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Byte 0	B2 state	Collective error fast shut-off	B1 state	Configuration state	A1 state	External module state	Internal module state	Reserved
Byte 1	Module state output data	Module state of input data	Reserved	Reserved	IQ3+IQ4 power requirement 0: Excess current 1: no excess current	IQ1+IQ2 power requirement 0: Excess current 1: no excess current	Q3+Q4 power requirement 0: Excess current 1: no excess current	Q1+Q2 power requirement 0: Excess current 1: no excess current
Byte 2	I8 vs. T2/4 test pulse comparison	I7 vs. T1/3 test pulse comparison	I6 vs. T2/4 test pulse comparison	I5 vs. T1/3 test pulse comparison	I4 vs. T2/4 test pulse comparison	I3 vs. T1/3 test pulse comparison	I2 vs. T2/4 test pulse comparison	I1 vs. T1/3 test pulse comparison
Byte 3	I16 vs. T2/4 test pulse comparison	I15 vs. T1/3 test pulse comparison	I14 vs. T2/4 test pulse comparison	I13 vs. T1/3 test pulse comparison	I12 vs. T2/4 test pulse comparison	I11 vs. T1/3 test pulse comparison	I10 vs. T2/4 test pulse comparison	I9 vs. T1/3 test pulse comparison
Byte 4	Reserved	Reserved	Reserved	Reserved	IQ4 vs. T2/4 test pulse comparison	IQ3 vs. T1/3 test pulse comparison	IQ2 vs. T2/4 test pulse comparison	IQ1 vs. T1/3 test pulse comparison
Byte 5	I15/I16 two-channel state 0: Error 1: ok or not used	I13/I14 two-channel state 0: Error 1: ok or not used	I11/I12 two-channel state 0: Error 1: ok or not used	I9/I10 two-channel state 0: Error 1: ok or not used	I7/I8 two-channel state 0: Error 1: ok or not used	I5/I6 two-channel state 0: Error 1: ok or not used	I3/I4 two-channel state 0: Error 1: ok or not used	I1/I2 two-channel state 0: Error 1: ok or not used

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Byte 6	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved	IQ3/IQ4 two-channel state 0: Error 1: ok or not used	IQ1/IQ2 two-channel state 0: Error 1: ok or not used
Byte 7	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved
Byte 8	Q4 Stuck at low	Q4 Stuck at high	Q3 Stuck at low	Q3 Stuck at high	Q2 Stuck at low	Q2 Stuck at high	Q1 Stuck at low	Q1 Stuck at high
Byte 9	IQ4 (Output) Stuck at low	IQ4 (Output) Stuck at high	IQ3 (Output) Stuck at low	IQ3 (Output) Stuck at high	IQ2 (Output) Stuck at low	IQ2 (Output) Stuck at high	IQ1 (Output) Stuck at low	IQ1 (Output) Stuck at high

Module state bits of the I/O modules SP-SDI and SP-SDIO

NOTICE The module state bits for the SP-SDI and SP-SDIO are only fully supported from firmware version 1.2.x.

The module state bits have the following meaning, if not otherwise indicated:

0 = error

1 = no error

Table 8: Meaning of module state bits of EA modules SP-SDI and SP-SDIO

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Byte 0	Reserved	Collective error fast shut-off	Power supply for Q1 .. Q4	Configuration of this module is valid.	Not used (error history flag)	External module state	Internal module state	Not used ("executing state")
Byte 1	Module state of output data	Module state of input data	Reserved	Reserved	Two-channel evaluation of input I7–I8	Two-channel evaluation of input I5–I6	Two-channel evaluation of input I3–I4	Two-channel evaluation of input I1–I2
Byte 2	Test impulse comparison I8 vs. X2	Test impulse comparison I7 vs. X1	Test impulse comparison I6 vs. X2	Test impulse comparison I5 vs. X1	Test impulse comparison I4 vs. X2	Test impulse comparison I3 vs. X1	Test impulse comparison I2 vs. X2	Test impulse comparison I1 vs. X1
Byte 3	Q4 Stuck-at low 0: Stuck-at error 1: no stuck-at	Q4 Stuck-at high 0: Stuck-at error 1: no stuck-at	Q3 Stuck-at low 0: Stuck-at error 1: no stuck-at	Q3 Stuck-at high 0: Stuck-at error 1: no stuck-at	Q2 Stuck-at low 0: Stuck-at error 1: no stuck-at	Q2 Stuck-at high 0: Stuck-at error 1: no stuck-at	Q1 Stuck-at low 0: Stuck-at error 1: no stuck-at	Q1 Stuck-at high 0: Stuck-at error 1: no stuck-at

Module state bits of the SP-DIO I/O module

The module state bits have the following meaning if not otherwise indicated; normally only the first byte of the total state is transmitted:

0 = error

1 = no error or reserved

Table 9: Meaning of module state bits of the SP-DIO expansion module

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Byte 0	Reserved	Reserved	Power supply Y1-Y4 and IY5-IY8	Configuration state	Not used (error history flag)	External module state	Internal module state	Not used ("executing state")
Byte 1	Module state output data	Module state input data	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved
Byte 2	Reserved							
Byte 3	Reserved							

Module state bit of the gateways

The module state bits have the following meaning if not otherwise indicated; normally only the first byte of the total state is transmitted:

0 = error

1 = no error

Table 10: Meaning of gateway module state bits

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Byte 0	Reserved	Module state output data	Module state input data	Configuration state	Not used (error history flag)	Reserved	Internal module state	Not used ("executing state")
Byte 1	Reserved							
Byte 2	Reserved							
Byte 3	Reserved							

Example

Module 2 (SP-SDIO) has a short-circuit after high (24 V) at output 3. The following module state is transmitted to the network (only the first 20 of 60 bytes are shown):

Byte address	00	01	02 ...	03 11	04 12	05 13	06 14	07 15	08 16	09 17	10 18	11 19	...
Byte	3 0	2 1	1 ...	0 11	3 0	2 1	1 2	0 3	3 0	2 1	1 2	0 3	...
Value	FF	FF	FF	FF	FF	FF	FF	FF	**EF FB**	FF	FF	**FB EF**	...
Meaning	State of Module 0 (CPU)				State of Module 1 (SDIO)				**State of Module 2 (SDIO)**				...

The first relevant byte for the module 2 error described above is module state byte 0 for module 2. This is byte 11 with the hexadecimal value FB (1111 1011):

Bit #	7	6	5	4	3	2	1	0
Value	1	1	1	1	1	**0**	1	1

This corresponds to the error message "Summary of bits 0.5 ibs 0.7 (external error)", byte 0, bit 2 in the following table: *Meaning of module state bits of EA modules SP-SDI and SP_SDIO [ch. 3.3.5, p. 20]*

The second relevant byte is the module state byte 3 for module 2. This is byte 08 with the hexadecimal value EF (1110 1111):

Bit #	7	6	5	4	3	2	1	0
Value	1	1	1	**0**	1	1	1	1

This corresponds to the error message "Short circuit monitoring of output 3, short circuit after high", byte 3, bit 4 in the following table: *Meaning of module state bits of EA modules SP-SDI and SP_SDIO [ch. 3.3.5, p. 20]*

NOTICE

- Reserved (for future use) = static 1 (no state change)
- Not used (can be 0 or 1), both values occur.
- If there is no module, all values - including the reserved values - are set to logical 1.

3.4 Data received from the network (network output data sets)

The data from data set 1 (max. 50 bytes) received from the network may be differently arranged, depending on the protocol. For the Modbus TCP, this data set was divided into five data blocks, each with 10 bytes. In the SP-PROFIBUS-DP, output data blocks 1-4 each contain 12 bytes, while output data block 5 contains 2 bytes. CANopen only defines 4 data blocks, each with 8 bytes.

Table 11: Output data block 1–5 of the various gateways

	Size of output data block				
Gateway	**Block 1**	**Block 2**	**Block 3**	**Block 4**	**Block 5**
SP-PROFIBUS-DP / PROFINET IO	12 bytes	12 bytes	12 bytes	12 bytes	2 bytes
SP-CANopen	8 bytes	8 bytes	8 bytes	8 bytes	–
SP-EN-ETC / Modbus TCP / EtherNet/IP	10 bytes	10 bytes	10 bytes	10 bytes	10 bytes

The content of the output data blocks can be provided in the logic editor of the samosPRO COMPACT CPU, as well as via a second gateway in the samosPRO COMPACT gateway within the samosPRO COMPACT system for a different network.

NOTICE

- In order to use network data in the logic editor or as input for another network, you must assign a tag name for each bit to be used.
- Bits without specific tag names will not be available in the logic editor or for routing via a second gateway. Detailed information about how to assign tag names for the data received may be found in the corresponding sections of the chapters on the various gateways.
- You can monitor current communication with the network with the aid of input data state bits for receiving data from the network and the output data state bit for transmitting data to the network in the logic editor. When the gateway detects a communication error, both the content of the data sets and the associated state bit are set to zero (logical 0).
- When all communication fails, the data of the output data sets and the input data state bit are set to zero (logical 0).
- When a connection is closed while others remain available, the LED MS or LED state will flash red/green for a total of 10 seconds and an entry will be made in the error log. In this case the state bits are not affected.

ATTENTION

Do not use the same output data block number for two different PLC connections or TCP/IP sockets!
The output data block of the Ethernet gateways can be described in parallel via all communication interfaces or TCP/IP sockets (e.g. Modbus TCP/IP and Ethernet TCP/IP) if they make use of the same output data block number. In this case the last message will always overwrite the data received earlier.

4 Installation and basic configuration

4.1 Installing/removing

4.1.1 Installing modules on standard rail

ATTENTION

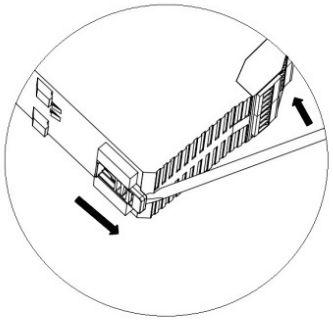
This is only for switchboxes with protection class IP 54 or higher!
The samosPRO system is only suitable for installations in a switchbox having at least protection class IP 54.

Info

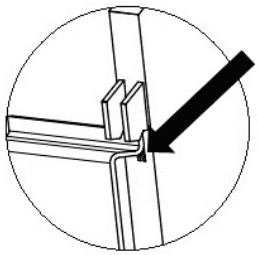
- ESD protection measures
  Note the suitable ESD protection measures during installation.
  Failure to do so could result in damage to the bus (internal safety bus).
- Protect connector openings
  Undertake suitable measures so that no foreign bodies can penetrate connector openings, particularly those for the program removable storage.
- Module width:
  The modules are placed in a mounting box that is 22.5 mm or 45 mm wide depending on type.
- Quality of standard rail
  The mounting boxes are suitable for 35-mm standard rails as per EN 60715.
- Sequence of modules:
  The SP-COPx controller module is inserted all the way to the left in a samosPRO system. The two optional gateways follow directly to the right next to the controller module.
- Save space for subsequent model replacement
  The modules are connected via the plug connection integrated into the housing. Note that the samosPRO modules must be pulled about 10 mm apart for a module replacement before the corresponding module can be removed from the standard rail.
- Standards to be considered
  Installation according to EN 50274

Procedure 1: Installation of main module SP-COPxxx

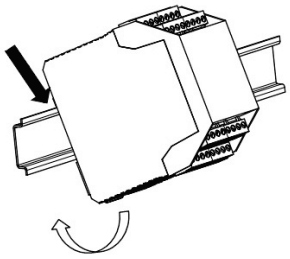
➡ Using a screwdriver, pull the snap-on foot outward.



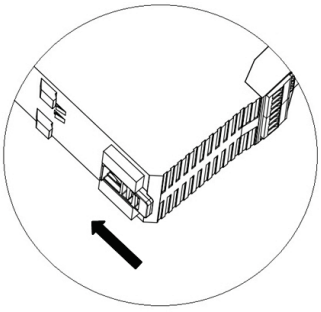
➡ Hang the module on the standard rail.
Important! Make sure that the grounding spring is seated correctly.
The grounding spring of the module must be placed on the standard rail so that it is secure and has good electrical conduction.



➡ Fold the module onto the standard rail.



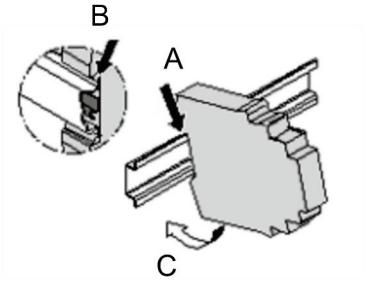
➡ Using a screwdriver, move the snap-on foot against the standard rail until the snap-on foot latches into position with an audible click.



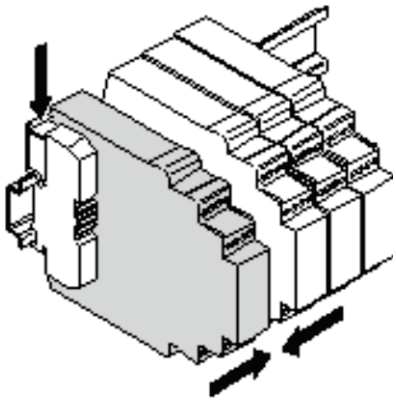
➡ Make sure that the module is securely seated on the standard rail.
Attempt to pull the module from the standard rail using slight pressure. If the module stays connected to the rail during this test, then the installation is correct.

Procedure 2: Installation of gateways or expansion modules

➡ Hang the module on the standard rail [A].
Important! Make sure that the grounding spring [B] is seated correctly.
The grounding spring of the module must be placed on the standard rail so that it is secure and has good electrical conduction.



➡ Using slight pressure, fold the module onto the rail in the direction of the arrow [C] until the module audibly latches into position.
➡ Make sure that the module is securely seated on the standard rail.
Attempt to pull the module from the standard rail using slight pressure. If the module stays connected to the rail during this test, then the installation is correct.
➡ If you are installing multiple modules:
Push the modules together individually in the direction of the arrow until the lateral plug connection between the modules audibly latches into position.



➡ Install an end cap into the module furthest to the left and another end cap into the module furthest to the right.

After installation

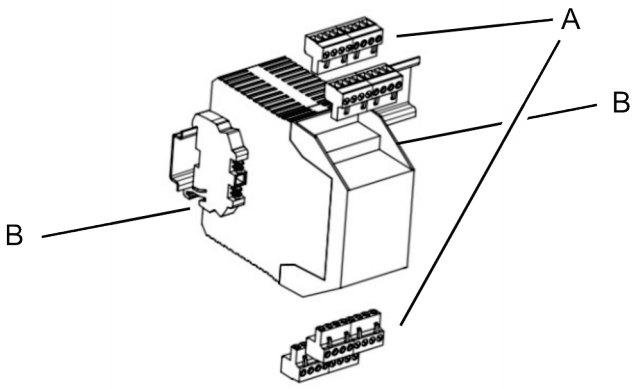
Once you have installed the modules, the following steps are required:

- Connect the modules electrically.
- Configure modules (see: "samosPLAN5+ Software" manual, BA000968).
- Check the installation before first commissioning.

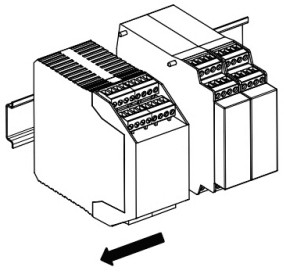
4.1.2 Removing modules from normal rail

Procedure 1: Removal of main module SP-COPxxx

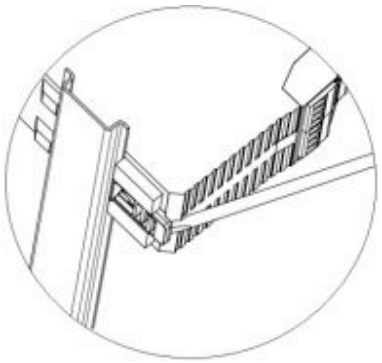
➡ Remove plug-in terminals with wiring [A] and remove the end terminals [B].



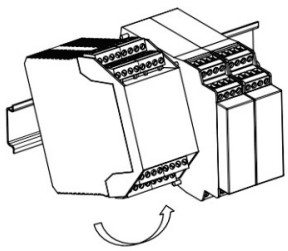
➡ With multiple modules:
Pull the modules apart from one another individually in the direction of the arrow until the lateral plug connection is disconnected.



➡ Release the modules.
To do this, pull the snap-on foot of the modules outward using a screwdriver.

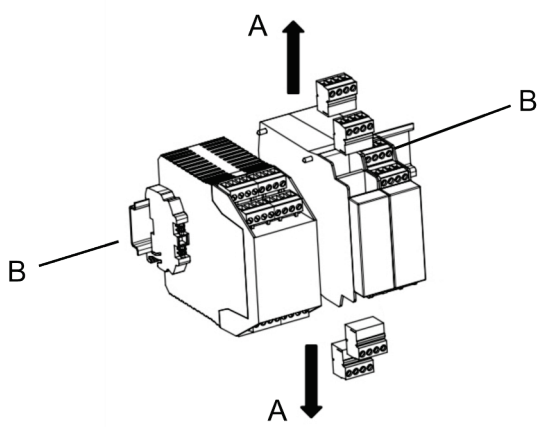


➡ Fold the module somewhat away from the standard rail and remove it from the rail.

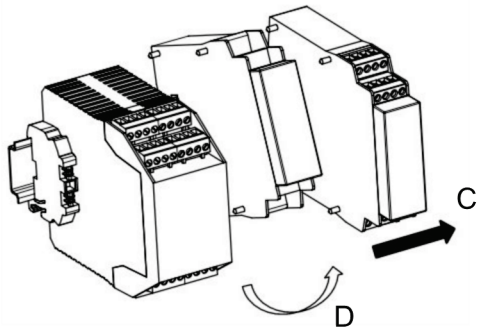


Procedure 2: Removing gateways or expansion modules

➡ Remove plug-in terminals with wiring [A] and remove the end terminals [B].



➡ With multiple modules:
Pull the modules apart from one another individually in the direction of the arrow [C] until the lateral plug connection is disconnected.



➡ Press on the module from above and fold the module away from the standard rail while it is in the pressed-down state [D].

4.2 Electrical installation

ATTENTION

Switch off the power supply to the system!
It is possible for the system to be unexpectedly started while you are connecting the devices.

NOTICE

- samosPRO COMPACT gateways meet EMC conditions as set out in the EN 61000-6-2 specification for use in an industrial environment.
- In order to ensure complete EMC safety, the standard rail must be connected to functional earth (FE).
- The switch box or installation housing for the samosPRO COMPACT system must meet at least the requirements of protection class IP 54.
- Installation according to EN 50274.
- Electrical installation as per EN 60204-1.
- The external power supply of the devices must be able to bridge a short-term power outage of 20 ms in accordance with EN 60204-1.
- The power supply must meet the regulations for low-voltage with safe disconnection (SELV, PELV) in accordance with EN 60664 and EN 50178 (equipping high-voltage systems with electronic equipment).
- Ensure that all modules of the samosPRO COMPACT system, the connected protective devices and the power supplies are connected to the same ground connection. The ground of the RS-232 interface is internally connected to the ground of the power supply for the main module (A2).
- Connect the shielding of all field bus and Ethernet cables to functional earth (FE) just before they lead into the switch box.

4.3 Initial configuration steps

How do you configure gateways? This chapter provides some brief guidelines.

Table 12: Guidelines for gateway configuration

Step	Description
1	**Establishing a link between the gateway and PC** See here for more detailed information: "samosPLAN5+ Software" (BA000968) manual, section "Connecting to the samosPRO system"
2	**Configure gateway** You will find detailed information in this regard at the following points in the gateway manual: • *Modbus TCP gateway [ch. 6, p. 47]* • *PROFINET IO-Gateway [ch. 7, p. 59]* • *EtherNet/IP gateway [ch. 8, p. 70]* • *PROFIBUS DP gateway [ch. 9, p. 107]* • *CANopen gateway [ch. 10, p. 118]* • *EtherCAT Gateway [ch. 11, p. 149]*
3	**Transmitting and verifying the configuration** See here for more detailed information: "samosPLAN5+ Software" (BA000968) manual, section "Transferring the system configuration"

5 Configuration of gateways with samosPLAN5+

The **Gateways** view in the samosPLAN5+ software has been provided for the configuration of gateways.

This section explains

- how the graphic user interface for the gateway configuration in samosPLAN5+ is structured,
- how you can carry out typical configuration tasks connected to gateways in samosPLAN5+.

NOTICE You will find more detailed information about the graphic user interface in the "samosPLAN5+ Software" manual (BA000968).

5.1 The graphic user interface

5.1.1 When the "Gateway" view is active

There are two ways of making use of the gateway function in samosPLAN5+. The **Gateway** view is only active, if you make use of one of these options in the **Hardware** view.

Scenario 1: You are using a gateway module

An SP-CAN module in the **Hardware** view has been selected in this example:

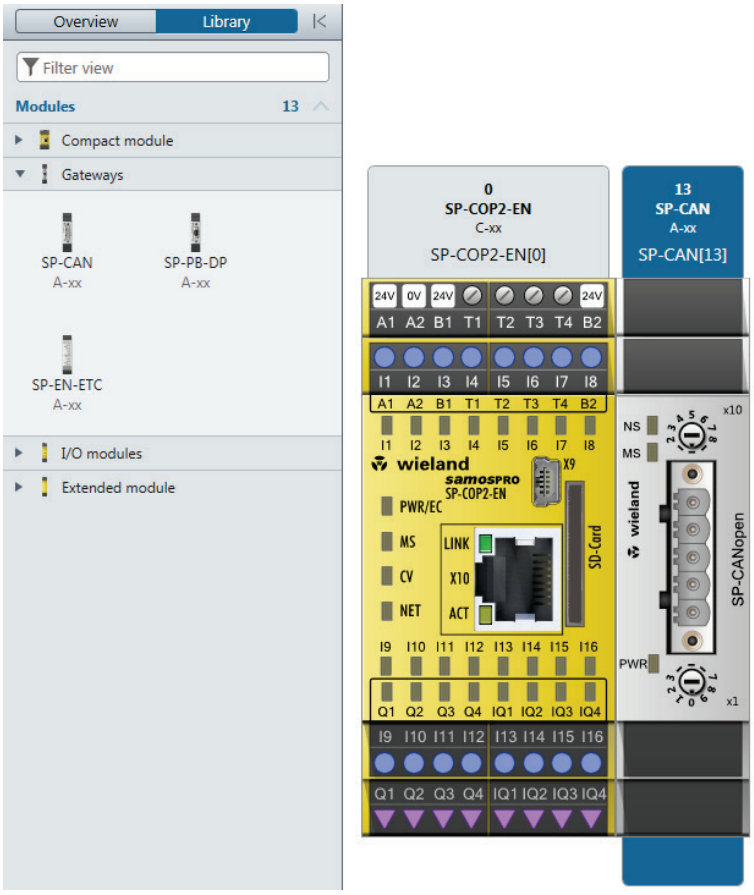


Illustration 1: Hardware configuration with gateway module

Scenario 2: You are using the gateway function on the "SP-COP2-ENI" module

The gateway function on the **SP-COP2-ENI** module can be adjusted n the right side bar, in the module configuration dialog (the module must first have been selected in the work area):

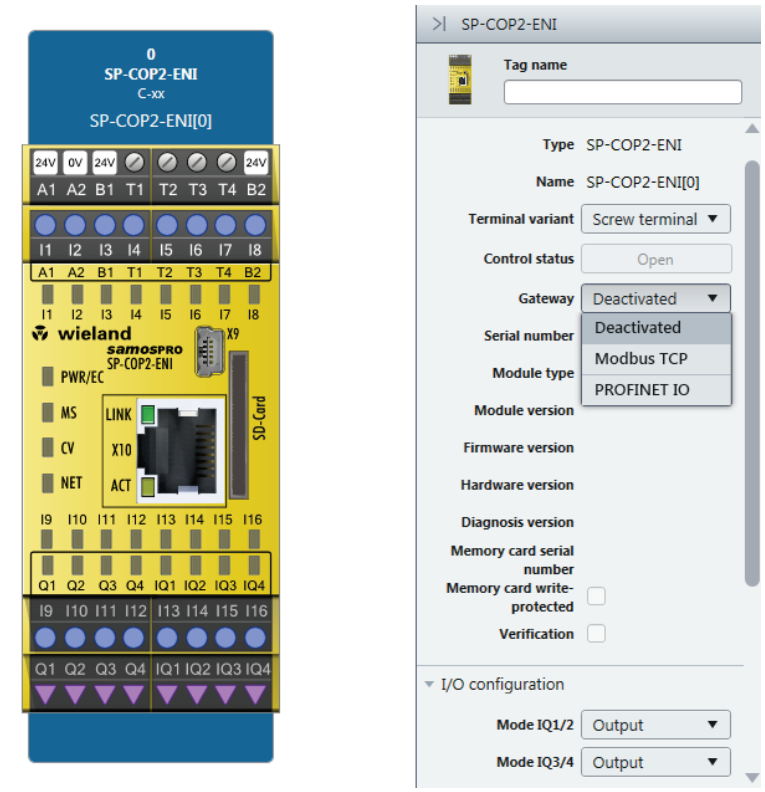


Illustration 2: "SP-COP-ENI" module with activated gateway function

5.1.2 Work area

Depending on the hardware configuration, you can see two or three tabs in the work area of the **Gateway** view. You can configure the gateways in these tabs.

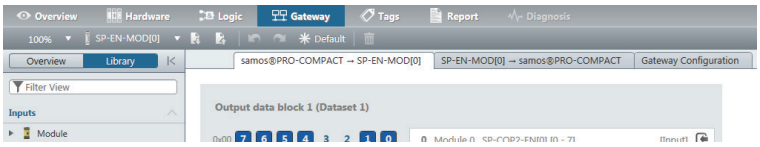
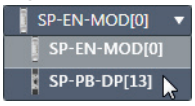


Illustration 3: Gateway configuration with three tabs

Display notes

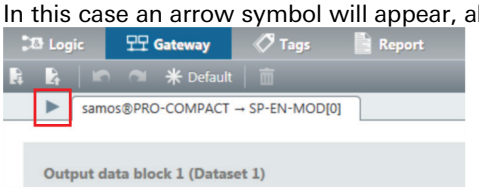
- **When you are using several gateways**

  The work area only ever shows a single gateway configuration. If you are using several gateways, you can toggle between the configurations by making use of the **Select data set view** menu:



- **When the program window is very small**

  If the window in which you have opened samosPLAN5+ is very small, not all tabs may be shown.
  In this case an arrow symbol will appear, allowing you to toggle between the tabs:



Tab 1: Routing table with output values (data bytes)

Transmission direction: samosPRO Compact -> network/field bus

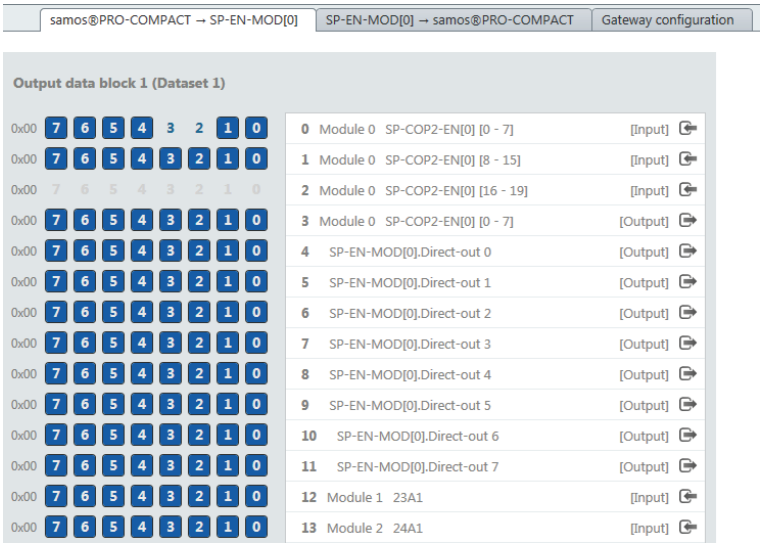


Illustration 4: Routing table with output values

Tab 2: Routing table with input values (data bytes)

Transmission direction: Network/field bus -> samosPRO Compact

When you are working with several gateways: This shows the mapping (the bits used are highlighted in blue), while the input data for the various gateways are shown in online mode (byte display 0x00 at the beginning of the relevant line).

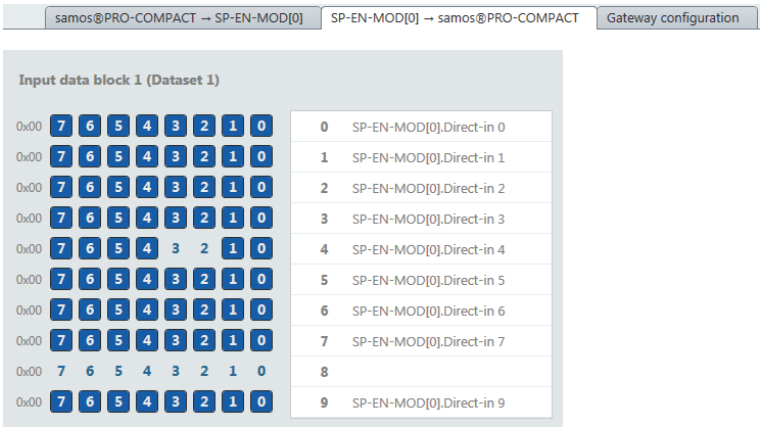


Illustration 5: Routing table with input values

Tab 3: "Gateway configuration"

Tab 3 only appears when you have activated the gateway function on the **SP-COP-ENI** module.

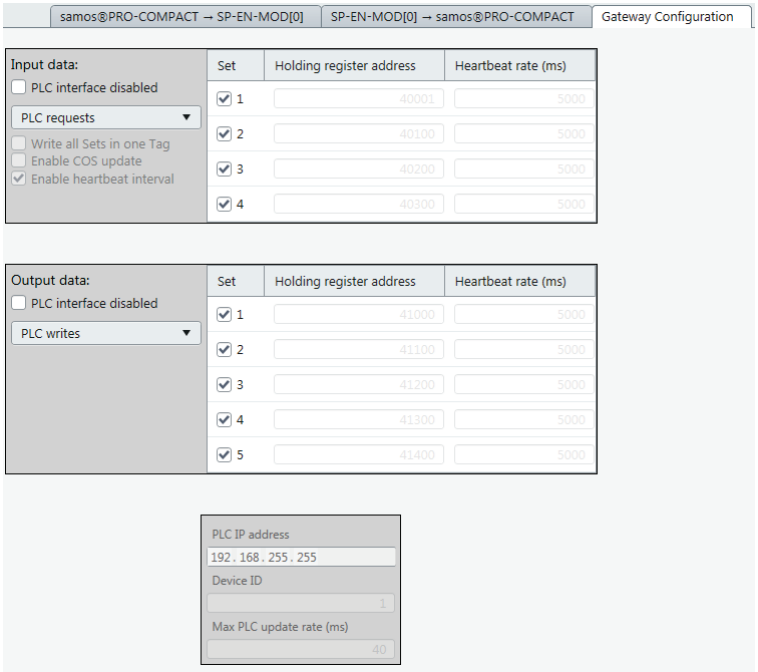


Illustration 6: "Gateway configuration" tab

NOTICE

Allocation of input and output data

The output and input data listed here refer directly to the data blocks in tab 1 and tab 2.

- **Output data** group **(to the PLC)**:
  Only **data set 1** can be configured. This refers directly to **output data block 1** in tab 1.
- **Input data** group **(to the PLC)**:
  Data set 1 to **data set 5** refer directly to **input data block 1** to **input data block 5** in tab 2.

5.1.3 Sidebars

Left sidebar | Library

The **Library** tab in the left side bar is only active when you have selected the first tab with the output data in the work area.

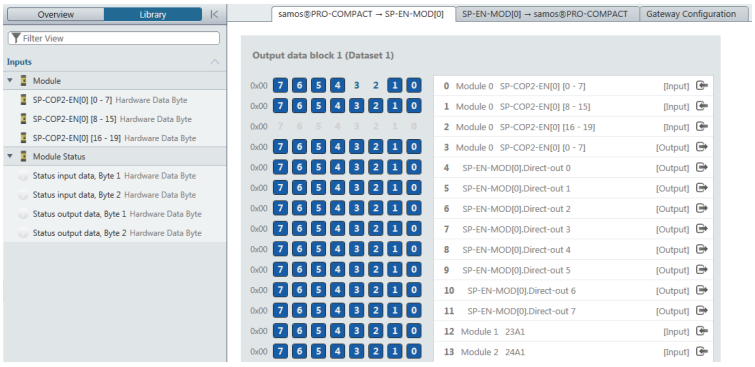


Illustration 7: Library in the "Gateway" view

Drag hardware data bytes from the library into empty fields in the routing table.

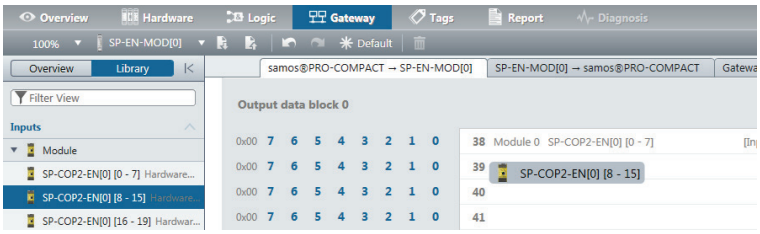


Illustration 8: Use drag&drop to move the data bytes to the routing table.

NOTICE You can make use of the same data byte several times in the routing table.

Left sidebar | Overview

In the **Overview** tab of the left sidebar, you can see all of the project components used as a hierarchical tree structure.

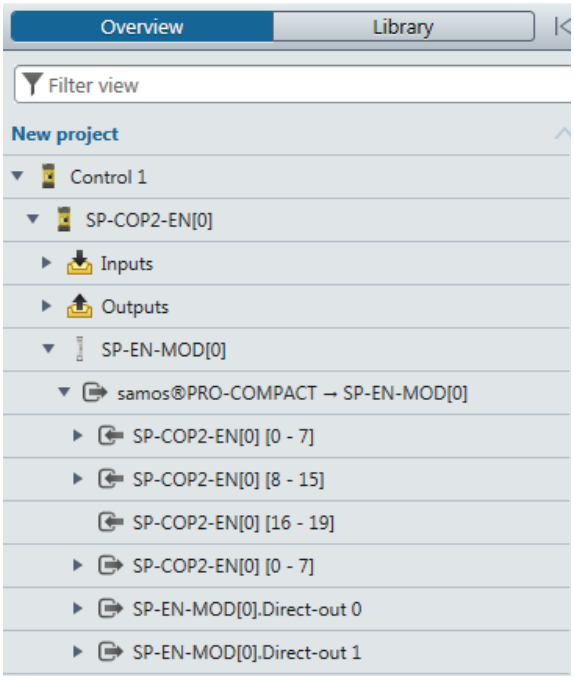


Illustration 9: Overview of the left sidebar

Right sidebar

The right side bar shows the configuration dialog for the data byte you have selected in the work area. Depending on the data byte, you can configure individual parameters.

You can also allocate tag names here.

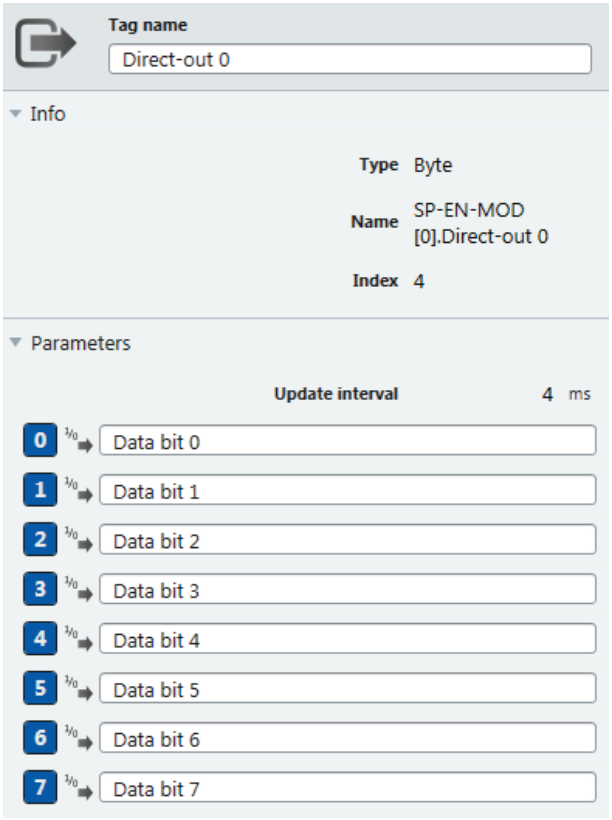
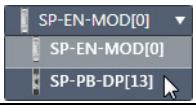
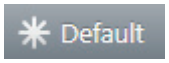


Illustration 10: Configuration dialog in the right sidebar

5.1.4 Commands

Via the command bar, you have access to the following view-specific functions:

Table 13: Key

Element	Description
	**Zoom** This determines the size of the display in the work area.
	**Data set view selection** When you are using several gateways: Changes between the gateway configurations.
	**Importing/exporting** Allows for the import/export of the configuration defined in the **Gateway** view. **Notes**: • **Caution:** When you import a configuration, all changes made before that have not been saved will be lost. You cannot undo this command. • Available storage formats: *SPG, *XML, *CSV You can use the import/export function to import the tag names used for a samosPRO COMPACT project into a PLC program or to export them from a PLC program into samosPLAN5+. • The **Import** button is only available for the routing configuration in the "network/field bus to gateway" direction.
	**Undo** This renders the last action undone.
	**Redo** This makes an action that has been undone redone.
	**Standard** This resets the configuration of the gateways to the basic settings. Also see: *Basic settings for the operating data [ch. 5.2.2, p. 38]*
	**Delete** This deletes the currently selected element.

5.2 Function and basic settings

5.2.1 Routing

The process diagram transferred to the network from the samosPRO COMPACT gateways comprises the operating data (e.g. logic results, state of inputs and outputs) and the diagnostic data (e.g. module state, CRCs.) These data have been arranged in 4 data sets.

Table 14: Content of data sets 1–4

Data set	Content	Size	Configurable
1	Operating data	50 bytes	Yes
2	CRCs	32 bytes	No
3	State and diagnosis	60 bytes	No
4	Reserved	60 bytes	No

The operating data in Data Set 1 may consist of up to 50 bytes, irrespective of the network protocol used. These 50 bytes have been divided into one or several data blocks, depending on the network protocol. Detailed information about the modularization of the data sent to the network may be found in the section on the relevant gateway and in the following table: *"Preset configuration for operating data transmitted to the network" [ch. 5.2.2, p. 38]*

The content of data set 1 has been preconfigured in the delivery state, but can be freely configured with a granularity of 1 byte (see *Basic settings for operating data [ch. 5.2.2, p. 38]* and *Configuring the gateway output values (tab 1) [ch. 5.3, p. 40]*).

The diagnostic data in data sets 2-4 depend on the network protocol used and are described in the chapter on the relevant gateway.

5.2.2 Basic settings for the operating data

The operating data have been preconfigured in the delivery state. Depending on the gateway used, these data are divided into several data blocks.

The following table provides an overview of which bytes have been allocated to the preset configuration and how the data at the various gateways are modularized.

Table 15: Preset configuration for operating data transmitted to the network

	Modbus TCP		PROFIBUS DP	
Byte	Preset allocation	Initial data set	Preset allocation	Initial data block
0	Input values for Module 0 (I1..I8)	#1	Input values for Module 0 (I1..I8)	#1
1	Input values for Module 0 (I9..I16)	(50 bytes)	Input values for Module 0 (I9..I16)	(12 bytes)
2	Input values for Module 0 (IQ1..IQ4)		Input values for Module 0 (IQ1..IQ4)	
3	Output values for Module 0 (Q1..Q4,IQ1-IQ4)		Output values for Module 0 (Q1..Q4,IQ1-IQ4)	
4	Direct data (Off) 0		Direct data (Off) 0	
5	Direct data (Off) 1		Direct data (Off) 1	
6	Direct data (Off) 2		Direct data (Off) 2	
7	Direct data (Off) 3		Direct data (Off) 3	
8	Direct data (Off) 4		Direct data (Off) 4	
9	Direct data (Off) 5		Direct data (Off) 5	
10	Direct data (Off) 6		Direct data (Off) 6	
11	Direct data (Off) 7		Direct data (Off) 7	
12	Inputs for Module 1		Inputs for Module 1	#2

	Modbus TCP		PROFIBUS DP	
Byte	Preset allocation	Initial data set	Preset allocation	Initial data block
13	Inputs for Module 2		Inputs for Module 2	(12 bytes)
14	Inputs for Module 3		Inputs for Module 3	
15	Inputs for Module 4		Inputs for Module 4	
16	Inputs for Module 5		Inputs for Module 5	
17	Inputs for Module 6		Inputs for Module 6	
18	Inputs for Module 7		Inputs for Module 7	
19	Inputs for Module 8		Inputs for Module 8	
20	Inputs for Module 9		Inputs for Module 9	
21	Inputs for Module 10		Inputs for Module 10	
22	Inputs for Module 11		Inputs for Module 11	
23	Inputs for Module 12		Inputs for Module 12	
24	Outputs for Module 1		Outputs for Module 1	#3
25	Outputs for Module 2		Outputs for Module 2	(12 bytes)
26	Outputs for Module 3		Outputs for Module 3	
27	Outputs for Module 4		Outputs for Module 4	
28	Outputs for Module 5		Outputs for Module 5	
29	Outputs for Module 6		Outputs for Module 6	
30	Outputs for Module 7		Outputs for Module 7	
31	Outputs for Module 8		Outputs for Module 8	
32	Outputs for Module 9		Outputs for Module 9	
33	Outputs for Module 10		Outputs for Module 10	
34	Outputs for Module 11		Outputs for Module 11	
35	Outputs for Module 12		Outputs for Module 12	
36-47	Not allocated		Not allocated	#4 (12 bytes)
48-49	Not allocated		Not allocated	#5 (2 bytes)

The preset allocation of the bytes can be freely configured, as shown in the following section.

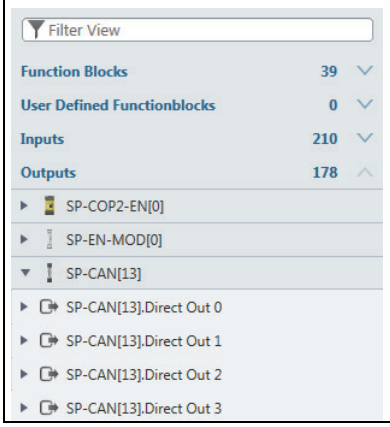
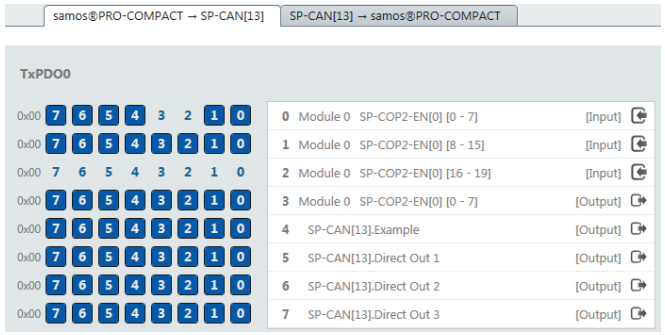
5.3 Configuring the gateway output values (tab 1)

You can use the following settings for the output values of a gateway in tab 1:

Basic setting

Depending on the gateway function selected, you will find four or eight bytes in tab 1, which are reserved as direct gateway output values. You can also see these bytes in the **Logic** view.

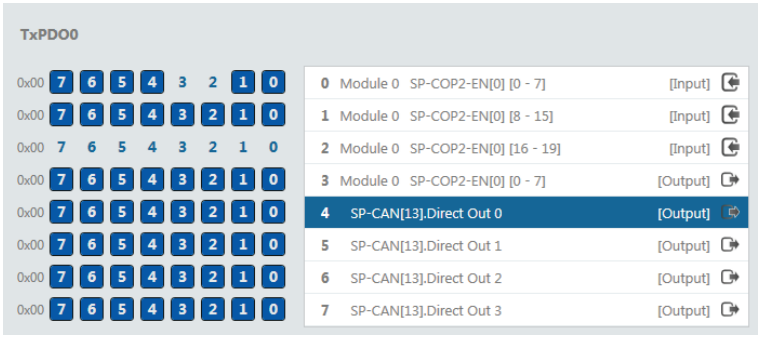
Example: **SP-CANopen** module with four predefined outputs for gateways:

"Logic" view	"Gateways" view
In the **Logic** view, you will see these four bytes in the left side bar under **Library | Outputs**:	In the **Gateways** view, these four outputs appear in the work area of the first tab:
	

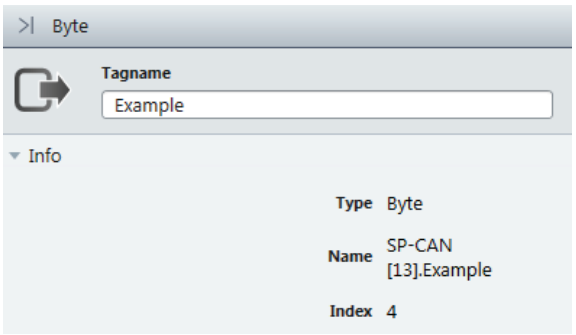
Change tag names of a predefined output value

Tag names have already been pre-assigned to the predefined output values (bytes). You can change these tag names:

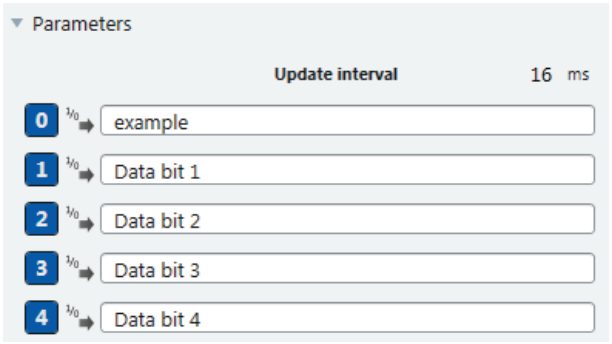
➡ In the work area, click on the byte for which you want to change the tag names.



➡ If you wish to change the tag name of the byte:
Overwrite the pre-allocated tag name of the byte with the desired new value in the configuration dialog.



➡ If you also want to change the tag names of individual bits:
Overwrite the pre-allocated values with the desired new value under **Parameters** in the configuration dialog.

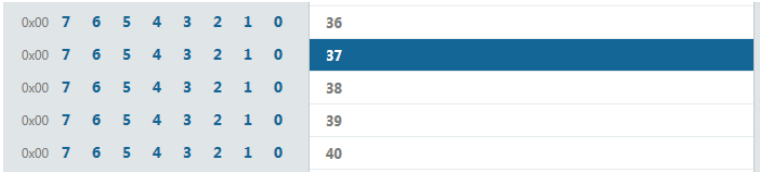


In the **Logic** view, these bits will appear with the corresponding tag names.

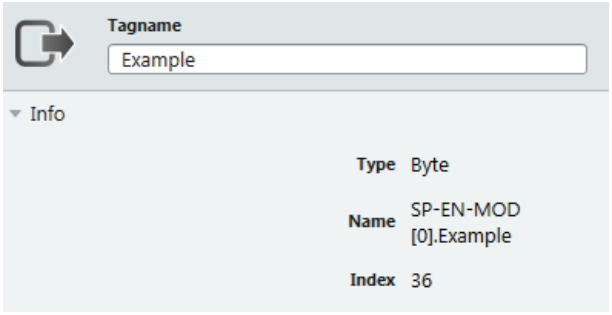
Configuring additional direct gateway output values

You can add new output values (bytes) in addition to the pre-allocated output values in the work area.

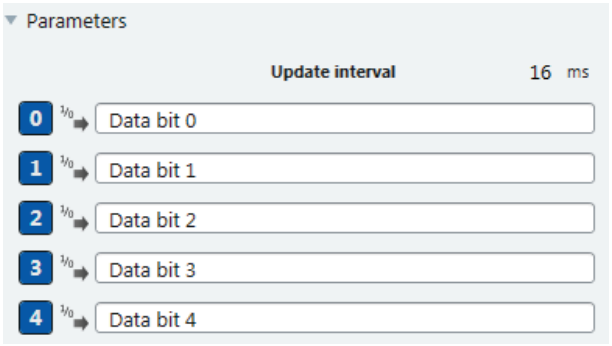
➡ Click on an empty byte in the work area.



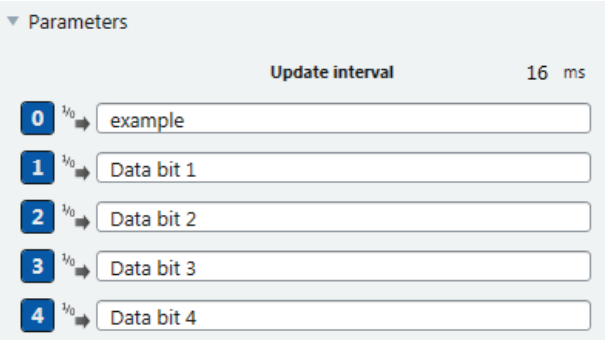
➡ Allocate a tag name to the byte in the right side bar.



⇨ Tag names for all bits are automatically pre-allocated under **Parameters**.



➡ If you also want to change the tag names of individual bits:
Overwrite the pre-allocated values with the desired new value under **Parameters** in the configuration dialog.



In the **Logic** view, these bits will appear with the corresponding tag names.

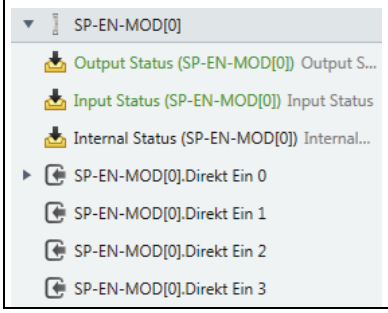
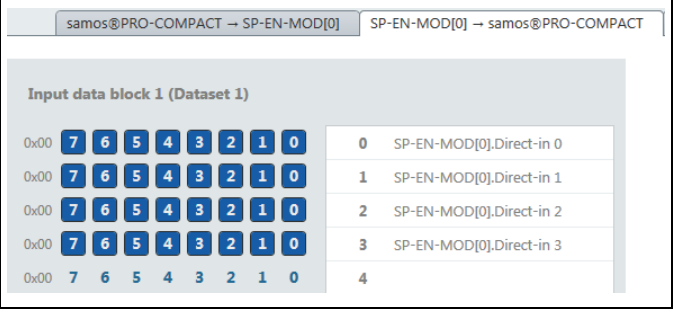
5.4 Editing the gateway input values (tab 2)

You can use the following settings for the output values of a gateway in tab 2:

Basic setting

Depending on the gateway function selected, you will find four or eight bytes in tab 2, which are reserved as direct gateway input values. You can also see these bytes in the **Logic** view.

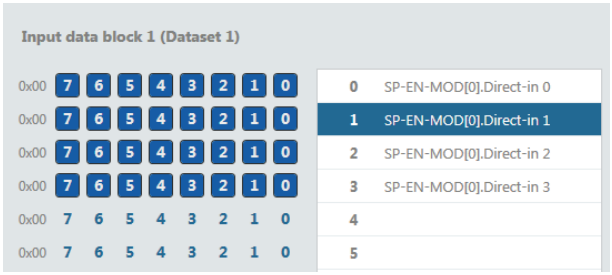
Example: **SP-CANopen** module with four predefined inputs for gateways:

"Logic" view	"Gateways" view
In the **Logic** view, you will see these four bytes in the left side bar under **Library | Inputs**:	In the **Gateways** view, these four inputs appear in the work area of tab 2:
	

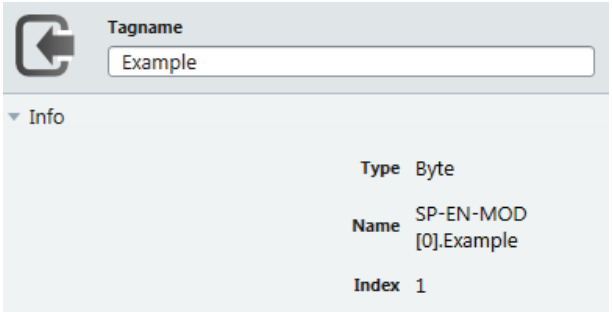
Change tag names of a predefined input value

Tag names have already been pre-assigned to the predefined input values (bytes). You can change these tag names:

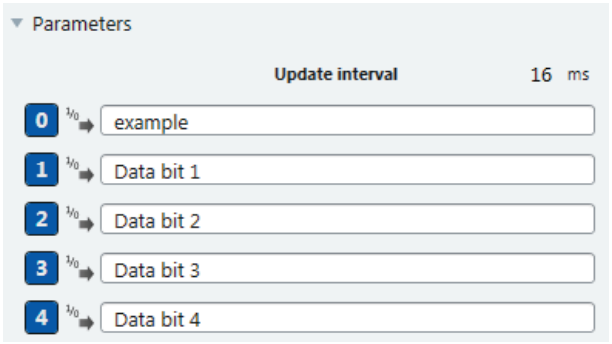
➡ In the work area, click on the byte for which you want to change the tag names.



➡ If you wish to change the tag name of the byte:
Overwrite the pre-allocated tag name of the byte with the desired new value in the configuration dialog.



➡ If you also want to change the tag names of individual bits:
Overwrite the pre-allocated values with the desired new value under **Parameters** in the configuration dialog.

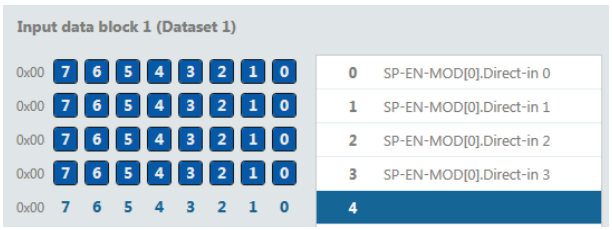


In the **Logic** view, these bits will appear with the corresponding tag names.

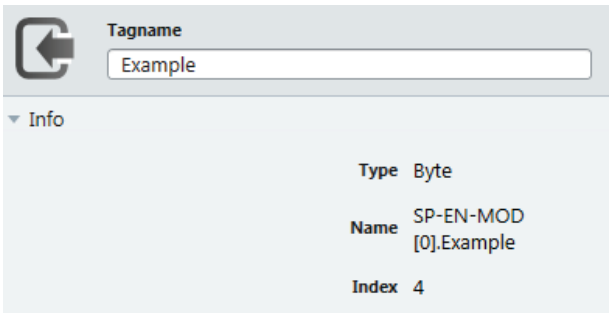
Configuring additional gateway input values

You can add new input values (bytes) in addition to the pre-allocated input values in the work area.

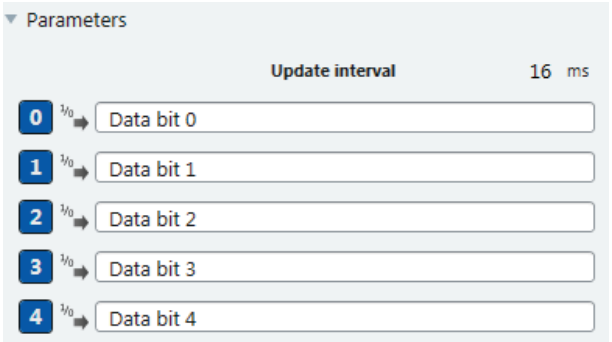
➡ Click on an empty byte in the work area.



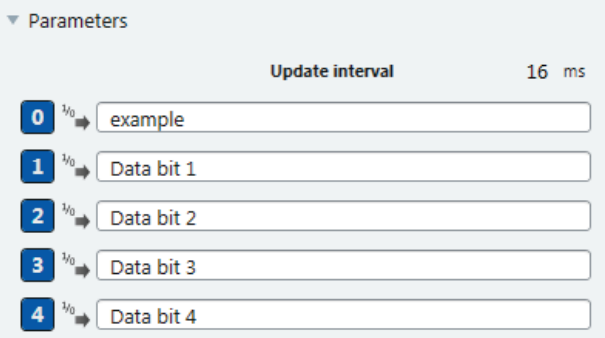
➡ Allocate a tag name to the byte in the right side bar.



⇨ Tag names for all bits are automatically pre-allocated under Parameters.



➡ If you also want to change the tag names of individual bits:
Overwrite the pre-allocated values with the desired new value under **Parameters**.



In the **Logic** view, these bits will appear with the corresponding tag names.

5.5 Monitoring operating data

You can monitor your gateway configuration directly in samosPLAN5+. This can be done in simulation mode (limited monitoring option) or by means of an active link to a samosPRO-System.

NOTICE

The samosPRO COMPACT gateways always show the actual physical state of the inputs and outputs of the connected modules and equipment. This means that even when the force mode is active and inputs that are physically **Low** are forced to **High** (or vice versa), the actual physical state of these inputs is transmitted to the PLC and not the (virtual) forced state. However, if one or several outputs change their state as a result of one or several inputs being forced, the changed state of these outputs will also be transmitted to the PLC, as the actual physical state of the equipment outputs has changed.

Simulation mode (offline mode)

You can test a gateway configuration offline in simulation mode. Use the logic analyzer for this purpose and manually set the desired inputs to **High** or **Low**.

Read here how to work with the simulation mode and logic analyzer: "samosPLAN5+ Software" (BA000968) manual, Section "Simulating configuration"

Monitoring with an active connection (online mode)

You can also test a gateway configuration online by establishing a link between samosPLAN5+ and a samosPRO system.

Read here how to activate the online mode and what you need to take into account: "samosPLAN5+ Software" (BA000968) manual, Section "Connecting to the samosPRO system"

NOTICE

LED behavior for active connections
If you are linked to a samosPRO installation in online mode, the state LEDs in the **Hardware** view of samosPLAN5+ will light up in the same way as for the connected system.
Further information about the state LED may be found in the documentation for the relevant module:

- *Modbus TCP gateway [ch. 6.4, p. 57]*
- *PROFIBUS-DB gateway [ch. 9.4, p. 116]*
- *CANopen gateway [ch. 10.13, p. 146]*

6 Modbus TCP gateway

The SP-COP2-ENI can be used for Modbus TCP. The internal SP-EN-MOD (Modbus TCP Gateway) is a component of the SP-COP2-ENI device and is activated by the gateway configuration:

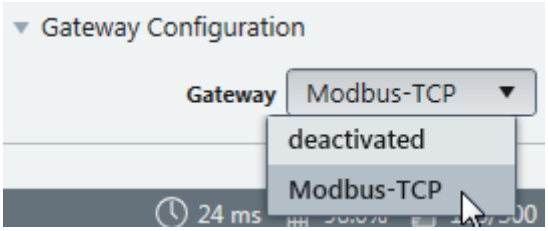


Illustration 11: Activation of Modbus TCP on the SP-COP2-ENI module

The samosPRO COMPACT Modbus TCP gateway supports the following:

- Modbus TCP with master and slave operation
- Ethernet TCP/IP socket interface, polling and auto-update function

6.1 Interfaces and operation

The SP-COP2-ENI is equipped with a RJ-45 socket.

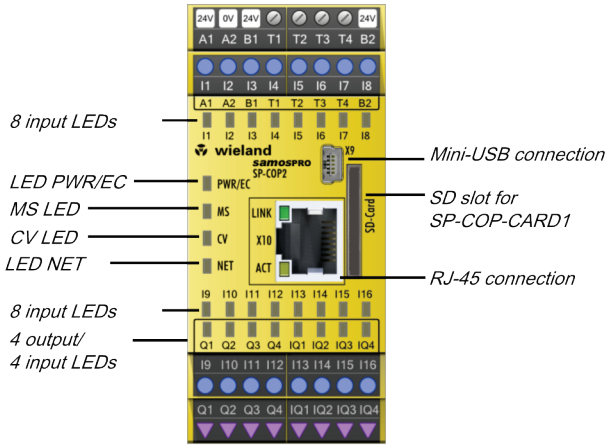


Illustration 12: Interfaces and display elements

Further information

- Here in this manual:
  Diagnostics and troubleshooting [ch. 6.4, p. 57]
- In the "samosPRO hardware" manual (BA000966):
  Device state and LED displays in the COMPACT modules (SP-COP1, SP-COP2-ENx)

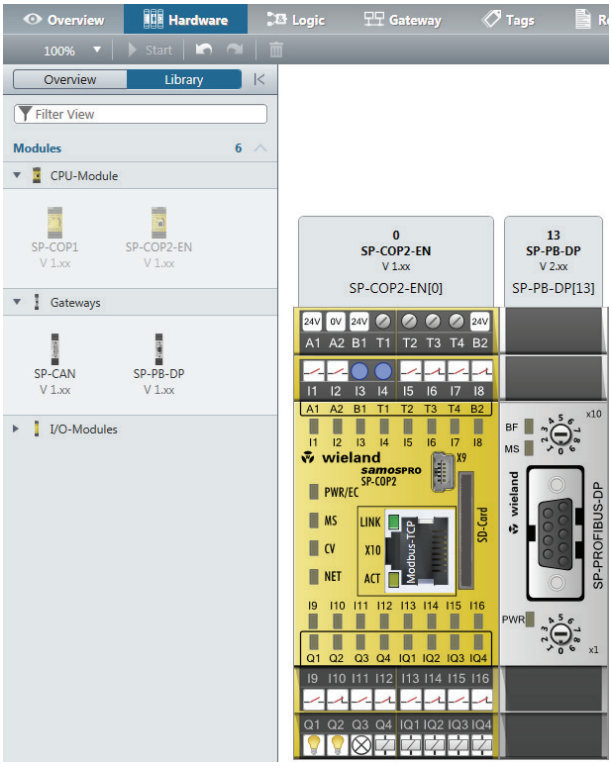
6.2 Basic configuration – allocation of an IP address

The SP- COP-ENI is configured with the aid of the samosPLAN5+ configuration software.

Procedure

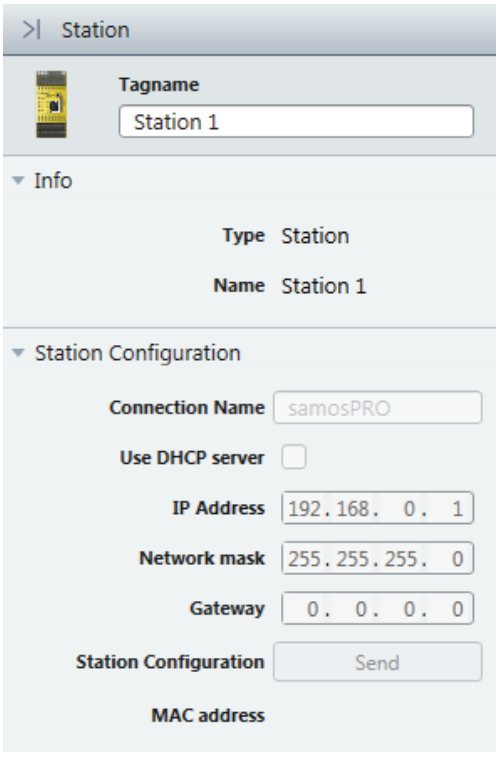
➡ Start samosPRO5+ and change to the **Hardware** view.

⇨ You can see the modules used in the work area.



➡ Use the mouse to click on the white background of the work area.

⇨ The configuration dialog of the controls will appear in the right side bar.



➡ Enter the following values under **Configuration of controls**:
- valid IP address
- subnet mask
- if required: valid IP address for a default gateway

OR:

Alternatively activate DHCP.

➡ Ensure that samosPLAN5+ is connected to the samosPRO system.
More detailed information on the link to the controls: "samosPLAN5+ Software" (BA000968) manual, Section "Connecting to the samosPRO system"

➡ Press **Send** in the right side bar to transfer the configuration to the samosPRO system.

6.3 Configuration of the Modbus-TCP interface to the PLC - how the data are transferred

Application characteristics for Modbus TCP:

- Support of standard addressing conventions for Modbus TCP.
- Master and slave operation

Requirements for the PLC for Modbus TCP:

- The PLC must support the Modbus TCP protocol.
- The PLC must either support the Read Holding Registers and Write Multiple Registers commands or the Read/Write Multiple Registers command.

The configuration steps in this section determine how the data are to be transmitted to the higher-level PLC.

There are two different methods of transmission for each transmission direction, i.e. samosPRO COMPACT **to network** and **network to** samosPRO COMPACT:

- Receiving method Polling/PLC requests (gateway as slave)

  This method allows the PLC regularly to request data using polling.

  When this method is used, the data are returned in the response to the data request. The PLC requests data by accessing the receiving data address of the SP-COP-ENI module via a read-holding-register telegram.
- The master receiving method gateway writes to the PLC (auto-update, gateway as master)

  When the SP-EN-MOD sends data to the PLC, these are immediately written to a memory location in the PLC.
- Slave transmission method - PLC writes (gateway as slave)

  With this method, the PLC sends telegrams to the SP-COP-ENI to write to the output data sets. For this purpose, the PLC writes data into defined addresses.
- The master transmission method gateway reads from the PLC (auto-update, gateway as master)

  With the master transmission method, the SP-COP-ENI polls the PLC for the output data sets.

NOTICE The configuration is regarded as faulty when the IP address of the PLC is zero and the read transfer mode and/or write transfer mode has been set for the master.

The number of possible connections to the PLC depends on whether the SP-COP-ENI is operated as a master or as a slave. Depending on the setting, up to 6 PLC can simultaneously address the SP-COP-ENI.

Table 16: Maximum number of possible Modbus TCP connections for the individual operating modes

Operating mode of the SP-COP-ENI	Maximum number of connections
Output data (to the PLC): Gateway writes Input data (from PLC): Gateway reads	1 outgoing connection 1 incoming connection
Output data (to the PLC): Gateway writes Input data (from PLC): PLC writes	1 outgoing connection 6 incoming connections
Output data (to the PLC): PLC reads Input data (from PLC): Gateway reads	6 outgoing connections 1 incoming connection
Output data (to the PLC): PLC reads Input data (from PLC): PLC writes	6 outgoing connections 6 incoming connections

The following table describes the configuration, depending on the transmission method:

Gateway is master

Table 17: Configuration directive – gateway as master

Essential settings in the gateway configuration (via samosPLAN5+)	Settings required for the PLC program and/or in the Modbus TCP configuration tool
Choose **Gateway writes to tag/file** and/or **Gateway reads from register** to configure the gateway as a master.	–
Select which data are to be written to the PLC or read from it.	–
Define where the selected data in the PLC memory are to be written to: Enter the register address(es). Example: "40001" and/or you can determine from which location in the PLC memory the selected data are to be read: Enter the register addresses.	Ensure that the addresses allocated in the samosPLAN5+ are available and that they contain the data intended for the samosPRO COMPACT system.
Choose how often these data are to be transmitted.	–
Define from and to where the data in the Modbus-TCP network are to be read and written: Enter the IP address and the slot number of the PLC controller.	–

Gateway as slave

Table 18: Configuration directive – gateway as slave

Essential settings in the gateway configuration (via samosPLAN5+)	Essential settings in the PLC program and/or in the Modbus-TCP configuration tool
Select **PLC requests** and **PLC writes** in the gateway configuration dialog.	–
–	Select which data are to be written to the gateway or read from it. Ensure that the PLC program writes the data into the addresses allocated to the gateway (see Table "*Data addressing for the SP-COP-ENI as the receiver [ch. 6.3, p. 56]*").

NOTICE The address settings for the Modbus TCP gateway are 1-based. Please add 1 to the register address set in the samosPLAN5+ for a 0-based address setting.

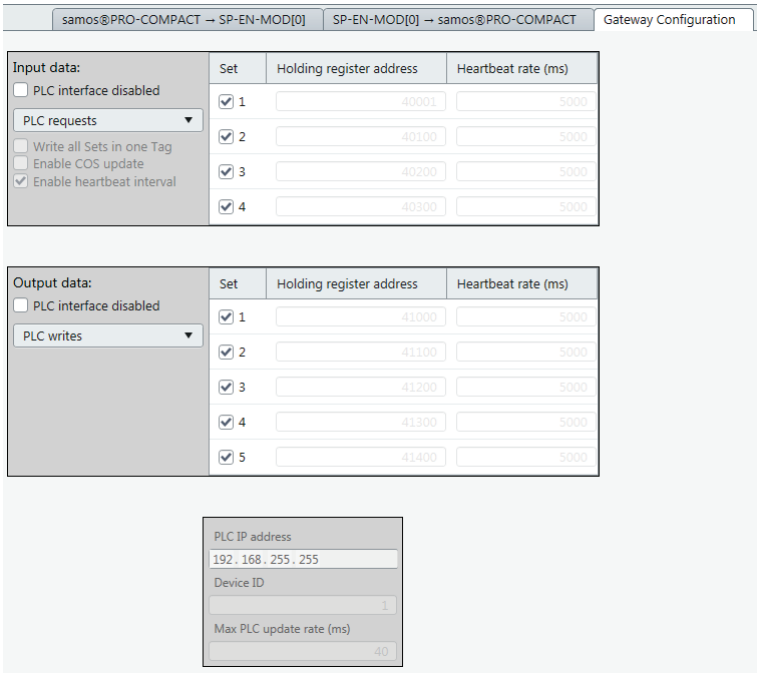
Master mode: SP-COP-ENI reads from/writes to the PLC

Carry out the following steps to configure the gateway as a master:

➡ Start samosPLAN5+.

➡ Change to the **Gateway** view and click on the **Gateway configuration** tab.

⇨ You will see the following configuration window:



➡ Choose the following settings:

Table 19: Settings for samosPRO Compact -> network direction

"Input data" area	
Selection list	Determines the transmission method. Value: **Gateway writes to tag/file**
Column **Set**	Determines which data are to be written to the PLC or read from it. Mark the control boxes for the desired data sets for this purpose. You will find a detailed description of the data sets here: *Data transferred to the network (network input data sets) [ch. 3.2, p. 14]*
Column **register address**	Define from and to where in the PLC memory the selected data should be read and written.
Column **heartbeat interval**	Defines how often the data sets are to be updated. Requirement: You have selected the option **Activate heartbeat interval** (see below).
All data sets in one tag	Optional Defines that all data sets are to be written to a single address in the PLC memory. In this case the register address defined for Data Set 1 will be used.

"Input data" area	
	Note: The following two settings can be simultaneously activated. They determine the frequency of data transmission.
Activating the COS update	Determines that the SP-COP-ENI immediately updates the data in the PLC as soon as changes are made to the data sets.
Activating the heart beat interval	Use the heartbeat intervals which you defined in the **Heartbeat interval** column to activate the update of the selected data sets.

Table 20: Settings for the network -> samosPRO Compact direction

"Output data" area	
Selection list	Determines the transmission method. Value: **Gateway reads from register**
Column **Set**	Determines which data are to be written to the PLC or read from it. Mark the control boxes for the desired data sets for this purpose. You will find a detailed description of the data sets here: *Data transferred to the network (network input data sets) [ch. 3.2, p. 14]*
Column **register address**	Define from and to where in the PLC memory the selected data should be read and written.
Column **heartbeat interval**	Defines how often the data sets are to be updated.

Table 21: Settings in Area 3

Area 3	
PLC IP address **Control ID**	The parameters define from and to where the data in the Modbus-TCP network are to be read and written:
Maximum refresh time for PLC	Define the maximum rate (or the minimum time interval) for transmitting the data sets to the PLC. This setting depends on the processing speed of the PLC. Minimum = 10 ms, maximum = 65535 ms. The basic setting of 40 ms is suitable for most PLC Note: When these values are greater than the heartbeat interval, the **heartbeat interval** will be slowed down to this value.

➡ Combine samosPLAN5+ with the samosPRO system and transmit the configuration. More detailed information on the link to the controls: "samosPLAN5+ Software" manual (BA000968), section "Combine with the samosPRO system"

Write to the PLC

NOTICE The following restrictions apply when the gateway operates as a master and writes the input data sets to the PLC:

- The address of the input data sets (preset in samosPLAN5+) must be the same as defined in the PLC.
- The PLC variable that is to incorporate the data must meet the following conditions:
  - in the address range 40xxxx (for Schneider Modicon PLC),
  - an array of 16-bit words,
  - long enough to contain the defined input data set array.
- All input data sets are transmitted to the PLC in 16-bit word format, with the first byte having the lowest value, i.e. on the far right of the integer, while the second byte has the highest value, i.e. on the very left of the integer.

Reading from the PLC

NOTICE The following restrictions apply when the gateway operates as a master and reads the output data sets from the PLC:

- The address of the output data sets must be the same as defined in the PLC.
- The PLC variable from which the data are requested must meet the following conditions:
  - They fall into the address range 40xxxx (for Schneider Modicon PLCs).
  - There is an array of 16-bit words for the output data sets that is long enough to accommodate the entire output data set.
- All output data sets are transmitted to the PLC in 16-bit word format, with the first byte having to be placed as the lowest value, i.e. on the far right of the integer, while the second byte will have the highest value, i.e. on the very left of the integer.

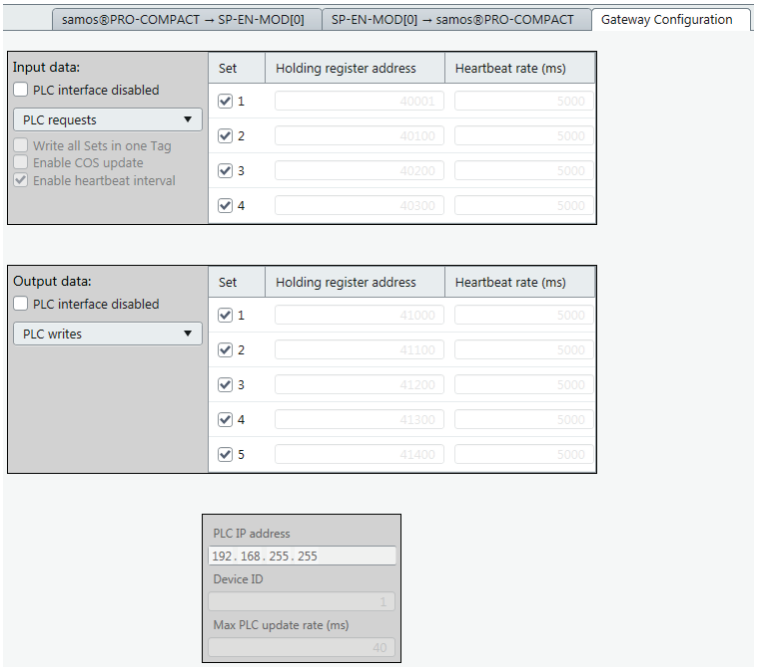
Slave mode - PLC reads from / writes to SP-COP-ENI

In this operating mode, the SP-COP-ENI sends the data as a slave at the request of the PLC. If this operating mode is desired:

➡ Start samosPLAN5+.

➡ Change to the **Gateway** view and click on the **Gateway configuration** tab.

⇨ You will see the following configuration window:



➡ Choose the following settings:

Table 22: "Input data" and "output data" area

Setting	Description/procedure
Selection list	Determines the transmission method. Enter the following settings for this purpose: • Input data: **PLC requests** • Output data: **PLC writes**
Data set column	Determines which data are to be written to the PLC or read from it. Mark the control boxes for the desired data sets for this purpose. You will find a detailed description of the data sets here: *Data transferred to the network (network input data sets) [ch. 3.2, p. 14]*

➡ Combine samosPLAN5+ with the samosPRO system and transmit the configuration. More detailed information on the link to the controls: "samosPLAN5+ Software" manual (BA000968), section "Combine with the samosPRO system"

PLC writes output data sets

The following restrictions apply when the PLC writes the output data sets:

- The equipment index must be 1.
- The telegram must be sent in Word format.
- All output data sets are transmitted to the PLC in 16-bit word format, with the first byte having to be placed as the lowest value, i.e. on the far right of the integer, while the second byte will have the highest value, i.e. on the very left of the integer.

PLC polls the input data sets

- The following restrictions apply:
- The equipment index must be 1.
- The PLC variable that is to incorporate the data must meet the following conditions:
  - It falls into the address range 40xxxx (for Schneider Modicon PLCs).
  - There is an array of 16-bit words that is long enough to accommodate the entire output data set.
- All input data sets are transmitted to the PLC in 16-bit word format, with the first byte having the lowest value, i.e. on the far right of the integer, while the second byte has the highest value, i.e. on the very left of the integer.

NOTICE Configure the PLC data polling in such a way that a data telegram is exchanged at least every minute between SP-COP2-ENI and the PLC. The TCP connection will otherwise be interpreted as not used and terminated.

NOTICE The data from the PLC to SP-COP2-ENI assume the value zero in the samosPLAN5+ logic program if the modbus TCP connection is terminated by the PLC itself or by a timeout.

SP-COP-ENI as slave – data addressing

The following table lists the addresses for reading out the data sets.

Unit ID 1

Table 23: Data addressing for the SP-COP-ENI as the receiver

Register (Base 1)	Description	Access	Scope (words)
1000	Request data for all activated input data sets	Reading	16-101 [1]
1100	Request data from input data block 1-5	Reading	25
1200	Request CRC data	Reading	16
1300	Request diagnostic data	Reading	30
1400	Reserved	Reading	30
2000	Write all activated output data sets	Write	5-25 [2]
2100	Write data from output data set 1	Write	5
2200	Write data from output data set 2	Write	5
2300	Write data from output data set 3	Write	5
2400	Write data from output data set 4	Write	5
2500	Write data from output data set 5	Write	5

[1] Corresponds to all activated input data sets.

[2] Must correspond to all activated output data sets. Example: If only output data sets 1 and 2 have been activated, 10 words (20 bytes) must be written. If all output data sets have been activated, 25 words (50 bytes) must be written.

Modbus commands and error messages

The SP-COP-ENI supports the following Modbus commands and error messages:

Table 24: Modbus commands

Modbus command	Value
Read holding registers	3
Read input[1] registers	4
Write multiple registers	16 (10hex)
Read/write multiple registers	23 (17hex)

[1] starting with module version A-03

Table 25: Modbus error messages

Modbus error response	Description
1 Function not permitted	The requested function is not supported
2 Data address not permitted	Undefined data address received
3 Data value not permitted	Request with prohibited data values, e.g. insufficient data requested for a data set
10 The gateway path is not available	Invalid configuration, e.g. polling or setting the digital outputs via PLC when operating the SP-COP-ENI in master mode

6.4 Diagnosis and troubleshooting

Information about the diagnosis of the samosPRO COMPACT system may be found in the "samosPLAN5+ Software" manual (BA000968).

Table 26: Troubleshooting for the SP-COP-ENI

Error	Possible cause	Possible remedy
Key: ○ LED off / ☀ LED flashes / ● LED lights up		
samosPLAN5+ cannot establish a link to the samosPRO COMPACT gateway	• The SP-COP-ENI has no power supply. • The SP-COP-ENI is not in the same physical network as the PC. • A different subnet mask has been set in the TCP/IP settings for the PC. • The SP-COP-ENI was been preconfigured and has a permanently set IP address or an IP address allocated to a DHCP server that has not been allocated.	• Switch on the power supply. Check the Ethernet wiring and the network settings of the P and correct them where necessary. • Set the PC to a network address 192.168.1.0 (The delivery state of the SP-COP-ENI has the address 192.168.1.4, which may not be used for the PC). • Alternatively activate DHCP on the PC and link SP-COP2-ENI and the PC to a network, using an active DHCP server. • Check the communication settings in samosPLAN5+.
The SP-COP-ENI is not providing any data. LED PWR/EC: ● Green LED LINK: ● Green LED ACT: ☀ Orange	• The SP-COP-ENI has been configured for data transmission to the PLC, but no Ethernet communication has been established or the communication is faulty. • Duplicate IP address detected. Another network device has the same IP address.	At least one Ethernet link must be established. • Set up the Ethernet link on the PC, check the Ethernet wiring, check the Ethernet settings in the PLC and in the samosPLAN5+. • If no Ethernet communication is required, deactivate the Ethernet connections / PLC interfaces on the SP-COP-ENI. • Correct the IP address and switch the system off and on again.

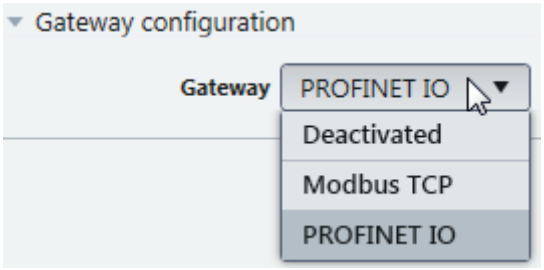
Error		Possible cause	Possible remedy
The SP-COP-ENI is not providing any data.		• Configuration required. • The configuration has not yet been fully transmitted. • The module version of the CPU does not support any samosPRO COMPACT gateways.	• Configure the SP-COP-ENI and transfer the configuration to the system. • Wait until the configuration has been fully transferred. • Use a CPU with the required module version (see Proper Use).
LED PWR/EC	Green		
LED LINK	Green		
LED ACT	Orange		
MS LED	Red (1 Hz)		
The SP-COP-ENI is not providing any data.		• No data set was activated. • No Ethernet communication interface was activated.	• Activate at least one data set.
LED PWR/EC	Green		
LED LINK	Green		
LED ACT	Orange		
MS LED	Green		
The SP-COP-ENI is not providing any data.		The SP-COP-ENI is in the Stop state.	The CPU/application has stopped. • Start the CPU (switch to Run mode) .
LED PWR/EC	Green		
MS LED	Green (1 Hz)		
The SP-COP-ENI functioned correctly after configuration, but has suddenly stopped providing any more data.		• The SP-COP-ENI is being operated in slave mode, the IP address is allocated by a DHCP server. • Following a restart of the SP-COP-ENI or the DHCP server, another IP address was allocated to the SP-COP-ENI, which was unknown to the PLC.	• Allocate a fixed IP address to the SP-COP-ENI. **or** • Reserve a fixed IP address for the SP-COP-ENI in the DHCP server (manual assignment using the MAC address of the SP-COP-ENI).
LED PWR/EC	Green		
LED LINK	Green		
LED ACT	Orange		
MS LED	Green		
The SP-COP-ENI / samosPRO COMPACT system is in the critical error state.		• The SP-COP-ENI is not properly connected to the other samosPRO COMPACT modules. • The module connection plug is dirty or damaged. • Another samosPRO COMPACT module has an internal critical error. • The voltage supply for SP-COP2-ENI is or was outside the specifications.	• Correctly plug in the SP-COP-ENI. • Clean the connection plug and socket. • Switch on the power supply again. • Check the power supply. • Check the other samosPRO COMPACT modules.
LED PWR/EC	Red		
LED LINK	Green		
LED ACT	Orange		
MS LED	Red		

7 PROFINET IO-Gateway

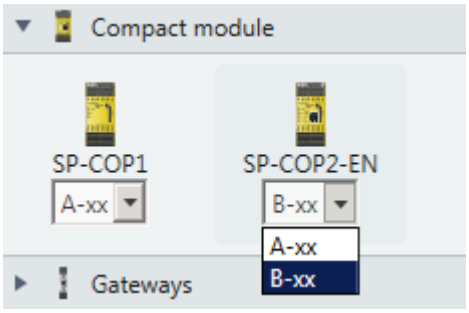
The SP-COP2-ENI can be used for PROFINET IO.

You will find the GSDML file and the equipment symbol for integration in a PLC of the product website of the SP-COP2-ENI on the Internet (http://eshop.wieland-electric.com).

The internal SP-EN-PN module (PROFINET IO Gateway) is a component of the SP-COP2-ENI device and is activated by the gateway configuration:



The selection of SP-EN-PN is possible for SP-COP2-ENI starting with module version B-xx:



The SP-COP2-ENI supports:

- PROFINET IO Conformance Class A
- Cyclical EA communication (RT)
- LLDP
- DCP
- Auto MDI
- Auto negotiation
- I&M 1-4
- Equipment diagnostics, alarms

Currently not supported:

- SNMP
- Shared Input, Shared Device
- FSU
- MIB II
- Port statistics

The number of PROFINET controllers (PLCs) which can simultaneous connect to the SP-COP2-ENI device via PROFINET is limited to one.

7.1 Interfaces and operation

Interfaces and operation are identical to that of the Modbus TCP Gateway.

Read the following section: *Interfaces and operation [ch. 6.1, p. 47]*

7.2 Basic configuration - Assigning a device name and an IP address

Configuration and diagnostics of the SP-COP2-ENI is possible both with the help of the samosPLAN5+ configuration software and the PROFINET IO network programming tool (e.g. SIEMENS TIA Portal).

Configuration using PROFINET IO

A MAC address is stored in every PROFINET IO field device such as the SP-COP2-ENI in the delivery state. The symbolic name (NameOfStation) **Test station** is stored on the SD card in the delivery state.

NOTICE

- In accordance with IEC 61158-6-10 no capital letters are permitted for the symbolic name (NameOfStation).
- This NameOfStation is used by the I/O controller (e.g. the PLC) to assign an IP address to the field device.
- If the IP address is also used for other communication via Ethernet, such as TCP/IP or for the configuration via Ethernet, please note that the PLC changes the IP address and can thus interrupt the other communication.

The IP address is assigned in two steps.

➡ Assign a unique system-specific name to the Gateway, using either the network configuration tool such as SIEMENS TIA Portal, or usamosPLAN5+. In samosPLAN5+, this is the connection name which can be edited in the hardware view.

➡ A (unique) system-specific name can be used by the I/O-Controller (i.e. the PLC) to assign the IP address to the gateway now before the system is booted.

NOTICE

The MAC address of the SP-COP2-ENI is printed on the device's nameplate (Example: 00:07:17:02:03:05).

Using the Siemens TIA Portal to assign device names

In the **Online accesses** area, select the network card connected to the network which can be used to access the SP-COP2-ENI device. In the **Assign name** function area, edit the **PROFINET device name** field and then select **Assign name**.

This will permanently assign the new device name to the SP-COP2-ENI device.

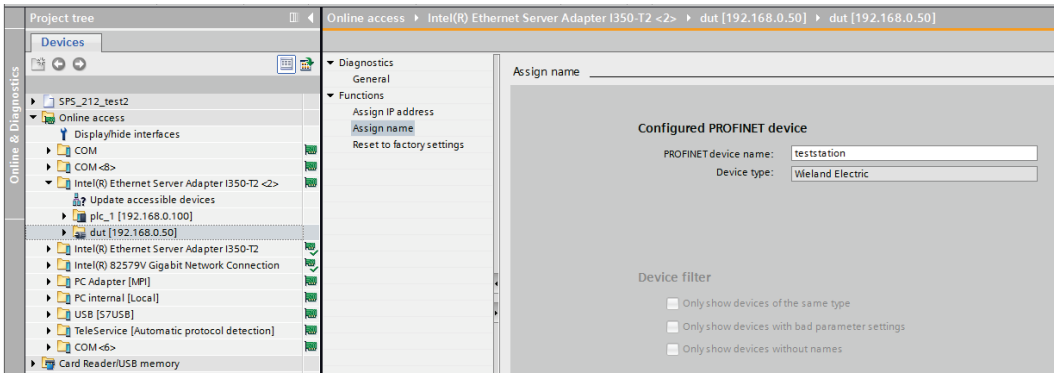


Illustration 13: Assigning device names with the TIA portal

Assigning device names via samosPLAN5+

- ➡ Start the samosPLAN5+ and connect it to SP-COP2-ENI.
- ➡ Press the **Stop** button in the **Hardware** view to stop the application.
- ➡ Click on the **Hardware** view in the blue area of the main window to open the right side of the controller configuration.
- ➡ Edit the connection name and click the **Send** button.

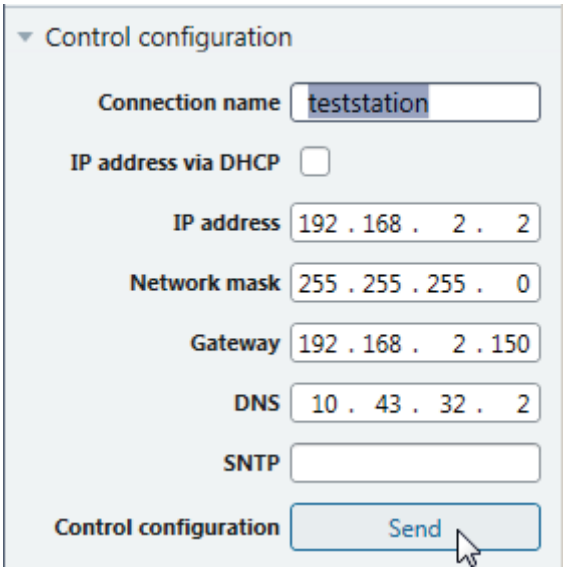


Illustration 14: Configuration dialog for the IP data and the device name

NOTICE

- The format of the device name must correspond to the specification of the PROFINET standard.
- Ensure that the address for the default gateway matches the address set by the PLC for the gateway. If no router is used, then Siemens Step 7 uses the same IP address for the default gateway as for the SP-COP2-ENI.
- If a project file with an active PROFINET IO is provided on SP-COP2-ENI then only one device in samosPLAN5+ can be found by USB. If you would like to use the Ethernet to connect with SP-COP2-ENI, then select **Edit** in the **Connect** dialog, where you then set the IP address of SP-COP2-ENI.

Use samosPLAN5+ to set the IP address

The IP address is typically assigned by the PROFINET IO controller (e.g. PLC). The SP-COP2-ENI, however, also allows the configuration of the entire samosPRO COMPACT system via Ethernet TCP/IP. It can be necessary in this case to already assign an IP address to the SP-COP2-ENI before the PROFINET IO network is set up. This can also be done in the configuration dialog shown above.

7.3 PROFINET configuration of the gateway - how the data are transferred

The following steps are required to configure the communication between the PLC and the gateway.

NOTICE

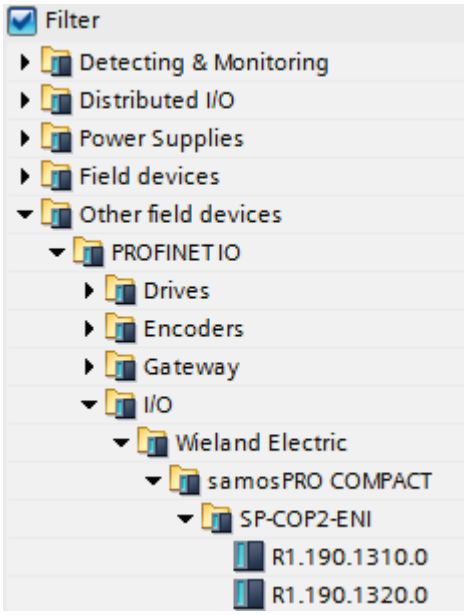
This documentation does not address the installation of the PROFINET IO network or the other components of the automation system project in the network configuration tool. It is assumed that the PROFINET project in the configuration program, e.g. the SIEMENS TIA Portal, has already been set up. The examples presented are based on configurations created with the help of the SIEMENS TIA Portal.

Step 1: Install the device master file (GDSML file)

Before the SP-COP2-ENI can be used for the first time as part of the network configuration tool, e.g. the SIEMENS TIA Portal, the device master file (GSDML file) of the gateway must first be installed in the hardware catalog of the tool.

- ➡ Download the GSDML file and the equipment symbol from the product site for the SP-COP2-ENI (http://eshop.wieland-electric.com).
- ➡ Follow the instructions for the installation of GSD files in the online help section or in the user manual for the PROFINET network configuration tool.

If you use the SIEMENS TIA Portal, then SP-COP2-ENI appears in the hardware catalog under **Additional field devices > PROFINET IO > I/O > Wieland Electric > samosPRO COMPACT > Head module > SP-COP2-ENI**



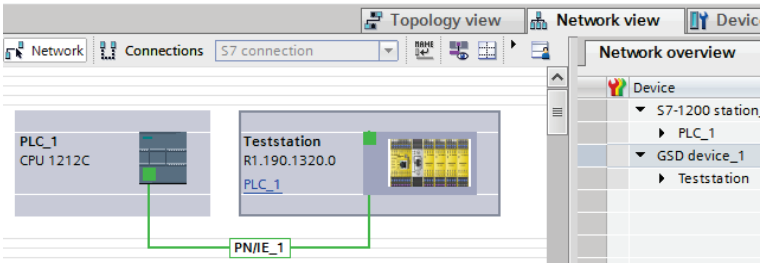
Step 2: Add the gateway to the project

To make the system data for the samosPRO COMPACT system available in the process diagram of the PLC, the gateway must first be added to the hardware configuration. The procedure to be used depends on the hardware configuration software of the PLC used. Please also read the documentation for the corresponding software in this regard.

The example below shows how the gateway is added to a SIEMENS TIA Portal project.

➡ Use Drag & Drop in the **Network view** to drag the device to the Ethernet PROFINET IO network.

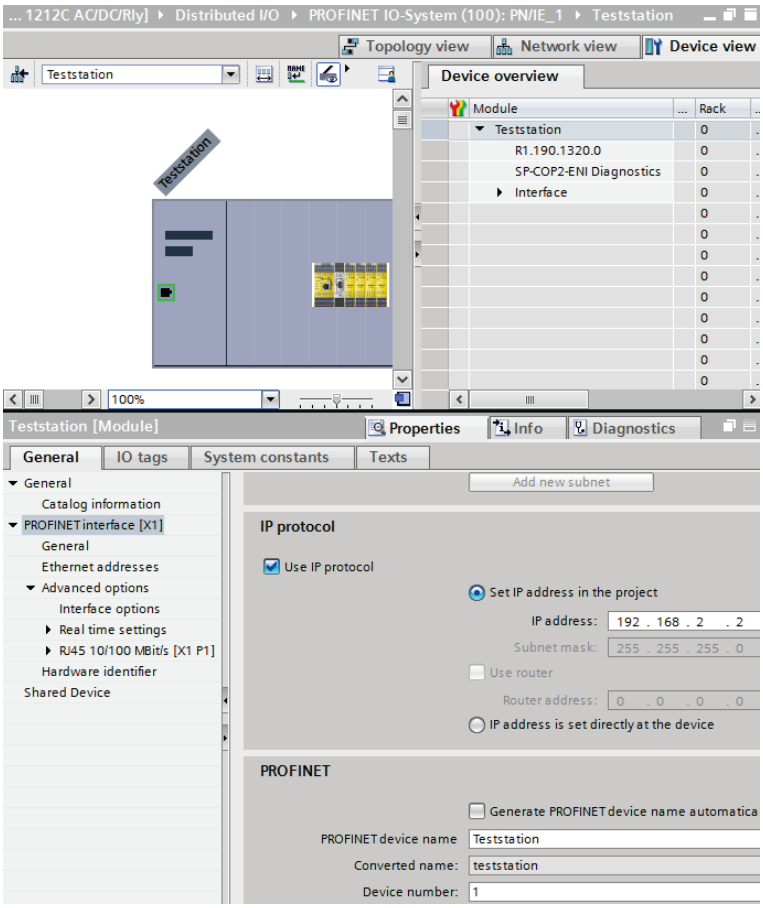
Example:



Step 3: Configure the gateway properties

➡ Double click on the SP-COP2-ENI hardware symbol.

➡ Configure the IP address, the device name, and the update interval of the cyclical I/O data exchange. Select the **Properties** tab for this.



NOTICE

The PLC can only communicate with the SP-COP2-ENI if the PLC software and the gateway use the same gateway name.
In accordance with IEC 61158-6-10 no capital letters are permitted for the symbolic name (NameOfStation).

7.4 PROFINET configuration of the gateway - which data are transferred

Cyclical data

The physical samosPRO-I/O modules are not presented in the PROFINET IO hardware catalog as typical hardware modules in the network. Instead, the data provided by the samosPRO COMPACT system have been arranged in various data blocks. Every data block represents a module in the PROFINET IO hardware catalog. The GSDML supports 13 Slots in which the modules can be placed. This makes is possible to use each data set one time (see illustration *"Configuration of the SP-COP2-ENI" [ch. 7.4, p. 65]*).

Process data from the samosPRO COMPACT system for the PLC

The SP-COP2-ENI provides 5 input data blocks (virtual device modules) which contain the process image. These can be exclusively placed in each corresponding slot 16 to 20.

NOTICE Input data blocks 1 to 4 each contain 12 bytes, while input data block 5 contains 2 bytes.

The content of the input data blocks can be freely selected. The data assignment in samosPLAN5+ is pre-configured in accordance with the following:

Table 27: Predefined content of input data block 1 to 5 of the SP-COP2-ENI

	Data block 1	Data block 2	Data block 3	Data block 4	Data block 5
Byte no. per data block	Input data	Input data	Input data	Input data	Input data
Byte 0	SP-COP2-ENI input values	I/O module 1 input values	I/O module 1 output values	Not allocated	Not allocated
Byte 1	SP-COP2-ENI input values	I/O module 2 input values	I/O module 2 output values	Not allocated	Not allocated
Byte 2	SP-COP2-ENI input values	I/O module 3 input values	I/O module 3 output values	Not allocated	Not available
Byte 3	SP-COP2-ENI output values	I/O module 4 input values	I/O module 4 output values	Not allocated	
Byte 4	Logic data values	I/O module 5 input values	I/O module 5 output values	Not allocated	
Byte 5	Logic data values	I/O module 6 input values	I/O module 6 output values	Not allocated	
Byte 6	Logic data values	I/O module 7 input values	I/O module 7 output values	Not allocated	
Byte 7	Logic data values	I/O module 8 input values	I/O module 8 output values	Not allocated	
Byte 8	Logic data values	I/O module 9 input values	I/O module 9 output values	Not allocated	
Byte 9	Logic data values	I/O module 10 input values	I/O module 10 output values	Not allocated	
Byte 10	Logic data values	I/O module 11 input values	I/O module 11 output values	Not allocated	
Byte 11	Logic data values	I/O module 12 input values	I/O module 12 output values	Not allocated	
Length	12 bytes	12 bytes	12 bytes	12 bytes	2 bytes
Byte offset	0	12	24	36	48

1 byte for data set 1 is available for every expansion module. The input values show the state of the preliminary evaluation of the I/O module. This corresponds to the state of the element in the main module logic. The level at the associated terminal cannot be clearly detected from this, as the data may be set to low, irrespectively of the level at the input terminal, by means of the cross-connection detection or two-channel evaluation (e.g. I1-18).

When two-channel input elements have been configured for an I/O module, only the lower-value bit represents the pre-evaluation state of the corresponding element (e.g. bit 0 for I1 and I2, bit 2 for I3 and I4, bit 4 for I5 and I6, bit 6 for I7 and I8).
The higher-value bit (bit 1, 3, 5 and 7) is used as follows in this case:
0 = error, 1 = no error

Further information

You will find information about how to configure the process diagram in the description of the *work area [ch. 5.1.2, p. 33]*.

Data from the PLC to the SP-COP2-ENI

There are 5 output data blocks having 10 bytes each. These can be exclusively placed in each corresponding slot 21 to 25.

The content of these data blocks can be used as input in the samosPLAN5+ logic editor or forwarded to another network by a second gateway. Every bit to be used must be assigned a tag name in order to provide the desired bits in the logic editor or for forwarding. Bits without tag names are not available.

Detailed information about how you can assign and adapt the tag names of the input and output data can be found here:
"samosPLAN5+ Software" manual (BA000968), Section"Assigning tag names / changing display names (exercise)"

NOTICE The standard value of the gateway data bit is zero following activation of the SP-COP2-ENI device.
If the connection to PLC is terminated, then all of the gateway data bits in the samosPLAN5+ logic program assume the value zero.

NOTICE For output data with IOPS=Bad, all of the gateway data bits in the samosPLAN5+ logic program assume the value zero. This is the case, for example, if the PLC is stopped.

Settings in the PROFINET IO network configuration tool

➡ Drag the required data blocks from the hardware catalog of the SIEMENS TIA Portal to the slots of the SP-COP2-ENI, found in the configuration table.

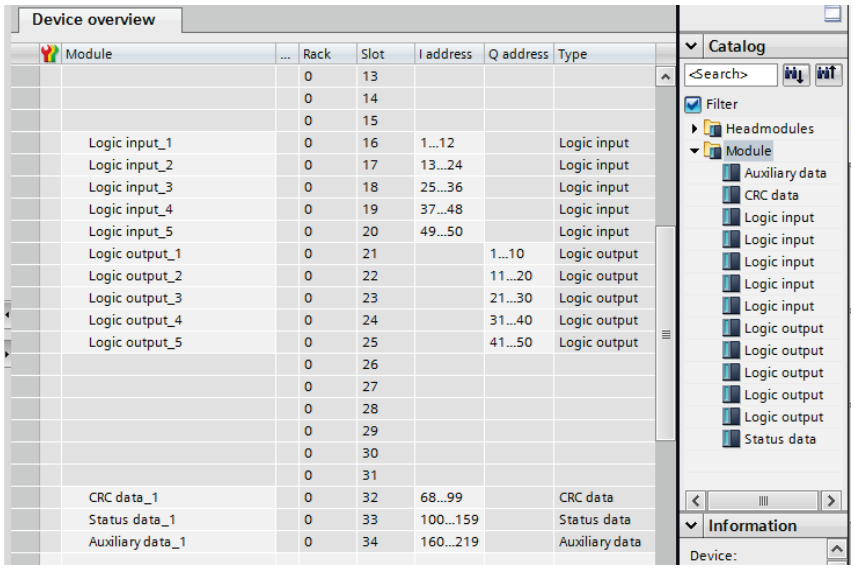


Illustration 15: Configuration of the SP-COP2-ENI

NOTICE The I and Q addresses indicate the location of the cyclical data in the memory.

Acyclical data and alarms

Read out data

The PLC can read out the diagnostic data of the samosPRO system. The diagnostic information is provided in three data sets, data sets 2, 3, and 4:

Data set 2 comprises 32 bytes and contains the project file's CRC 32. This can only be placed in slot 32.

Data set 3 comprises 60 bytes and contains the status of SP-COP2-ENI and the individual I/O modules. This can only be placed in slot 33. See the following to interpret the status bits in data set 3: Table *"Meaning of the module status bits of the main module" [ch. 3.3.5, p. 19]* and Table *"Meaning of the module status bits of the I/O modules" [ch. 3.3.5, p. 20]*

Data set 4 (auxiliary data) comprises 60 bytes and is currently filled with reserved values. This can only be placed in slot 34.

Information & Management

The SP-COP2-ENI supports the I&M information defined in the PROFINET IO specification. The following I&M information can be read out:

Table 28: Readable I&M information

Name	Size	Value range	I&M	Storage location
MANUFACTURER_ID (Vendor ID)	2 bytes	397 = 0x18D	0	SP-COP2-ENI
ORDER_ID (Order ID)	64 bytes	"R1.190.1310.0" + 51 blank spaces and "R1.190.1320.0" + 51 blank spaces	0	SP-COP2-ENI
SERIAL_NUMBER (IM_Serial_Number)	8 bytes	"16010001" to "99129999"	0	SP-COP2-ENI
HARDWARE_REVISION (IM_Hardware_Revision)	2 bytes	101 to 9999	0	SP-COP2-ENI
SOFTWARE_REVISION (IM_Software_Revision)	6 to 9 Bytes	"V0.1.0" to "V99.99.99"	0	SP-COP2-ENI
Device ID		1320	0	SP-COP2-ENI
REV_COUNTER (IM_Revision_Counter)	2 bytes	0 to 65535	0	SD card
PROFILE_ID (IM_Profile_ID)	2 bytes	0x0000 (Non-profile)	0	SP-COP2-ENI
PROFILE_SPECIFIC_TYPE (IM_Profile_Specific_Type)	2 bytes	0x0003 (IO modules)	0	SP-COP2-ENI
IM_VERSION (IM_Version)	2 bytes	1	0	SP-COP2-ENI
IM_SUPPORTED (IM_Supported)	2 bytes	10 (= 0b1010)	0	SP-COP2-ENI
TAG_FUNCTION	32 bytes	32 Bytes à 0x20..0x7E	1	SD card
TAG_LOCATION	22 bytes	32 Bytes à 0x20..0x7E	1	SD card
INSTALLATION_DATE (IM_Date)	16 bytes		2	SD card[1)]
DESCRIPTOR (IM_Descriptor)	54 bytes	54 Byte à 0x00..0xFF	3	SD card
IM_Signature	54 bytes	54 Byte à 0x00..0xFF	4	SD card
[1)] Subject to changes				

Alarms

Alarms can be acyclically read using the PROFINET IO alarm infrastructure. When an error in samosPRO COMPACT occurs, the PROFINET IO gateway sends a corresponding diagnostics alarm to the network. The details of the diagnostics alarm (text and help) are then available through the SIMATIC PLC interface. The RALRM (SFB54) function block in OB82 (diagnostics interrupt) allows you to make the details of the sent alarm directly available in the PLC program.

NOTICE

All alarms are output to module 0.
The cause of the alarm is displayed by an error message from the GSDML file.
The possible causes of an alarm can be found in the "samosPLAN5+ Software" manual (BA000968), Section "List of all error messages"

7.5 Diagnostics and troubleshooting

Information on the diagnostics of the samosPRO COMPACT system can be found in the "samosPLAN5+ Software" manual (BA0009678, Section "List of all error messages"

Table 29: Troubleshooting for the SP-COP2-ENI

Error		Possible cause	Possible remedy
Key: ○ LED off / ☀ LED flashes / ● LED lights up			
The SP-COP2-ENI is not providing any data.		• The SP-COP2-ENI has been configured for data transmission to the PLC, but no Ethernet communication has been established or the communication is faulty. • Duplicate IP address detected. Another network device has the same IP address. • Incorrectly formatted PROFINET device name	• PROFINET IO must be activated in the project file. At least one Ethernet link must be established. Check the Ethernet wiring, check the Ethernet settings in the PLC and in the samosPLAN5+. • Correct the IP address and switch the system off and on again. • Compare the device name between the PROFINET master and the SP-COP2-ENI.
LED PWR/EC	● Green		
LED LINK	● Green		
LED /ACT	☀ Yellow		
MS LED	● Green		
The SP-COP2-ENI is not providing any data.		• Configuration required. • The configuration has not yet been fully transmitted. • The module version does not support any PROFINET IO.	• Configure the SP-COP2-ENI with a project file in which PROFINET IO is activated and transfer the configuration to SP-COP2-ENI. • Use an SP-COP2-ENI device starting with module version B-xx.
LED PWR/EC	● Green		
LED LINK	● Green		
LED /ACT	☀ Yellow		
MS LED	☀ / ☀ Red/green		
The SP-COP2-ENI is not providing any data.		• The samosPRO system is in the stop state.	• Start the CPU (change to Run mode).
LED PWR	● Green		
LED LINK	● Green		
LED /ACT	☀ Yellow		
MS LED	☀ Green (1 Hz)		
The SP-COP2-ENI is not providing any data.		• The IP address for SP-COP2-ENI is assigned by a DHCP server. Following a restart of the SP-COP2-ENI or the DHCP server, another IP address was allocated to the SP-COP2-ENI, which is unknown to the PLC.	• Either assign the SP-COP2-ENI a permanent IP address or reserve a permanent IP address for the the SP-COP2-ENI in the DHCP server (manual assignment using the MAC address of the SP-COP2-ENI).
LED PWR/EC	● Green		
LED LINK	● Green		
LED /ACT	☀ Yellow		
MS LED	● Green		

Error		Possible cause	Possible remedy
The SP-COP2-ENI/ samosPRO system is in the critical error state.		• The SP-COP2-ENI is not properly connected to the other samosPRO COMPACT modules. • The module connection plug is dirty or damaged. • Another samosPRO module has an internal critical error.	• Insert the I/O module correctly. Clean the connection plug and socket. • Switch on the power supply again. • Check the other samosPRO modules.
LED PWR	Green		
LED LINK	Green		
LED /ACT	Yellow		
MS LED	Red		

7.6 Deactivation of the PROFINET IO function

If the SP-COP2-ENI device is started with an activated PROFINET IO function, then this function remains active until the device is switched off.

For this reason, switch the device off after sending a project without PROFINET IO function. This is required, for example, if you convert the gateway function in the samosPLAN5+ project from PROFINET IO to Modbus TCP.

8 EtherNet/IP gateway

This chapter describes the functionality of the EtherNet/IP gateway function of SP-COP2-ENI.

This chapter does not describe the protocol EtherNet/IP. If the reader is not familiar with this, he is referred to several document provided by the ODVA. The glossary lists some of it (see *Abbreviations and Definitions [ch. 8.10, p. 105]*).

8.1 Interfaces and operation

Interfaces and operation are identical to that of the Modbus TCP Gateway.

Read the following section: *Interfaces and operation [ch. 6.1, p. 47]*

8.2 Datasheet

The device SP-COP2-ENI supports EtherNet/IP as for product version D-01.01. Following features are implemented:

- Support of Implicit Messaging (Transport Class 1)
- Support of Explicit Messaging (Transport Class 3, connected)
- Meets the profile: General Purpose Discrete I/O Device
- UCMM (unconnected) Message Server
- Supported Objects: Message Router, Connection Manager, Port, Identity, Ethernet Link, TCP/IP, Discrete I/O Point & Group, Vendor Class 0x78, Assembly
- Up to 5 concurrent Encapsulation sessions (input & output)
- Variable Size Assemblies
- Supported addressing: Class/Instance/Attribute as well as Symbol Tag
- Conformance with CIP (Common Industrial Protocol) Specification, edition 3.18 and EtherNet/IP Adaptation of CIP, edition 1.19
- Detailed EDS-File with ODVA conformance test
- Supported PCCC commands: Word Range Read & Write, Typed Read & Write, Protected Typed Logical Read & Write with 2 and 3 Address Fields each to get connected also with PLC 3, PLC 5, PLC 5/250, PLC 5/VME, SLC 500, SLC 5/03, SLC 5/04 and MicroLogix-1000
- Auto-negotiation of Half- and Full-Duplex as well as 10 and 100 MBits/sec
- MS (Module State) and NET (Network) LED

8.3 Basic setup

8.3.1 Basic configuration of PLC

This chapter is an initial and quick installation guide. First, install the most current EDS file for the SP-COP2-ENI in your PLC configuration tool. You will find the most current EDS file on the Internet at http://eshop.wieland-electric.com. The following pictures illustrate a typical setup using Logix Designer.

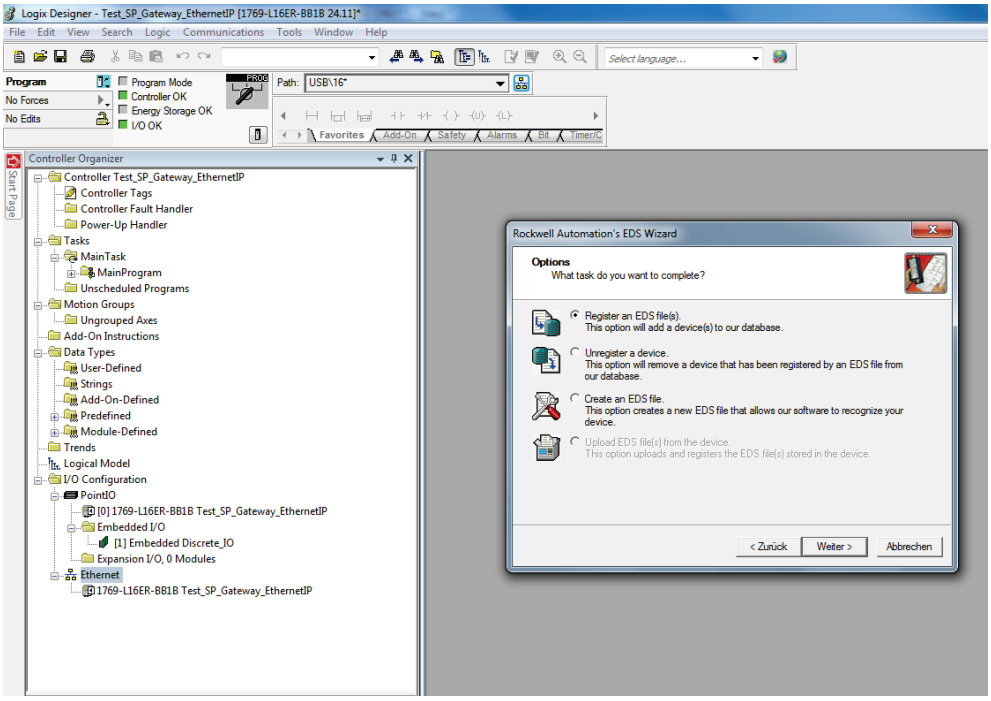


Illustration 16: Register EDS file in Logix Designer via EDS Wizard

The catalog number is R1.190.1320.0 which can be filtered by vendor name **Wieland Electric** or a part of this name.

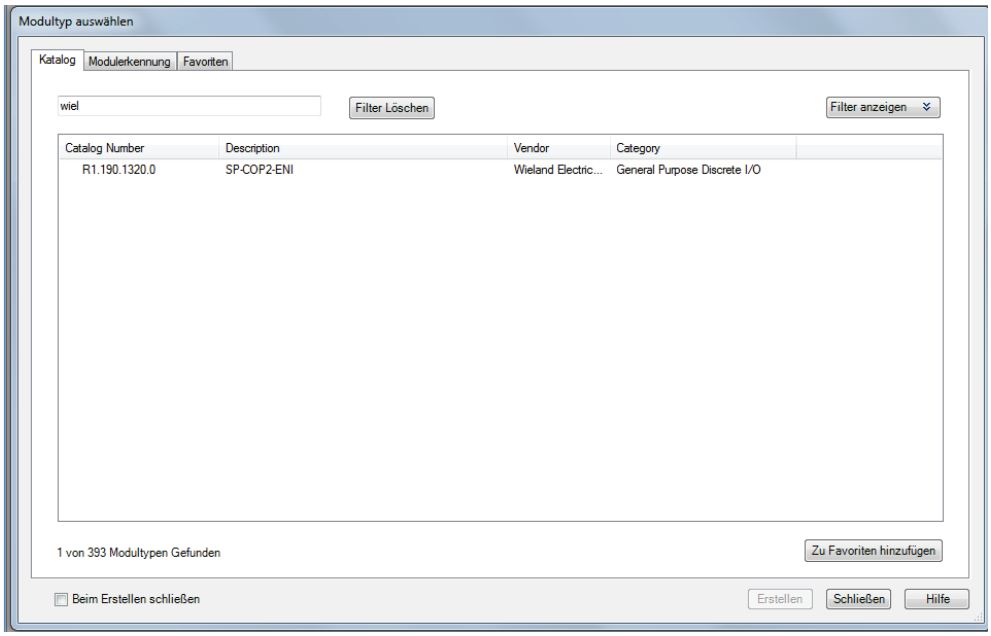


Illustration 17: Choosing module type in Logix Designer

In the tab **Internet Protocol**, select **Manually configure IP settings** in Logix Designer. Choose the proper IPv4 address and network mask.

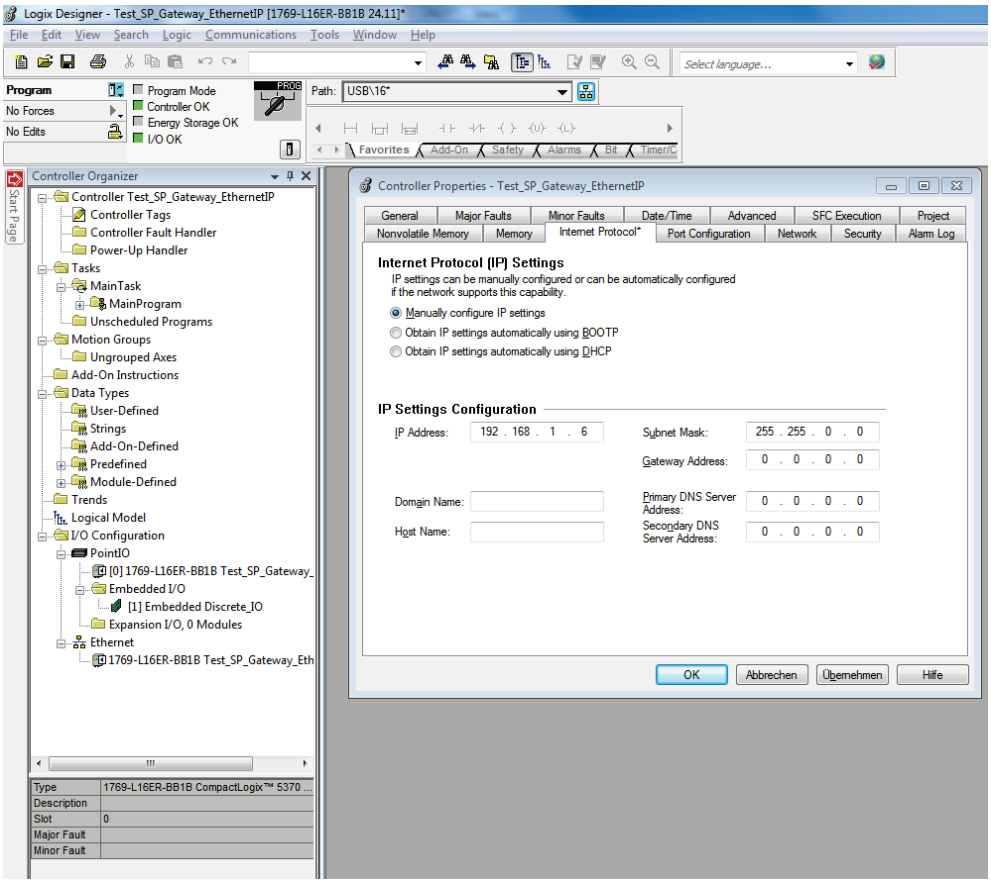


Illustration 18: IPv4 setup for device in Logix Designer

The device SP-COP2-ENI meets the device profile **General Purpose Discrete I/O Device**. For quick setup, connection **Logic Output (1 to 400) and Logic/Physical Input** can be used if your PLC supports implicit messaging. The following picture shows the appropriate dialog in Logix Designer.

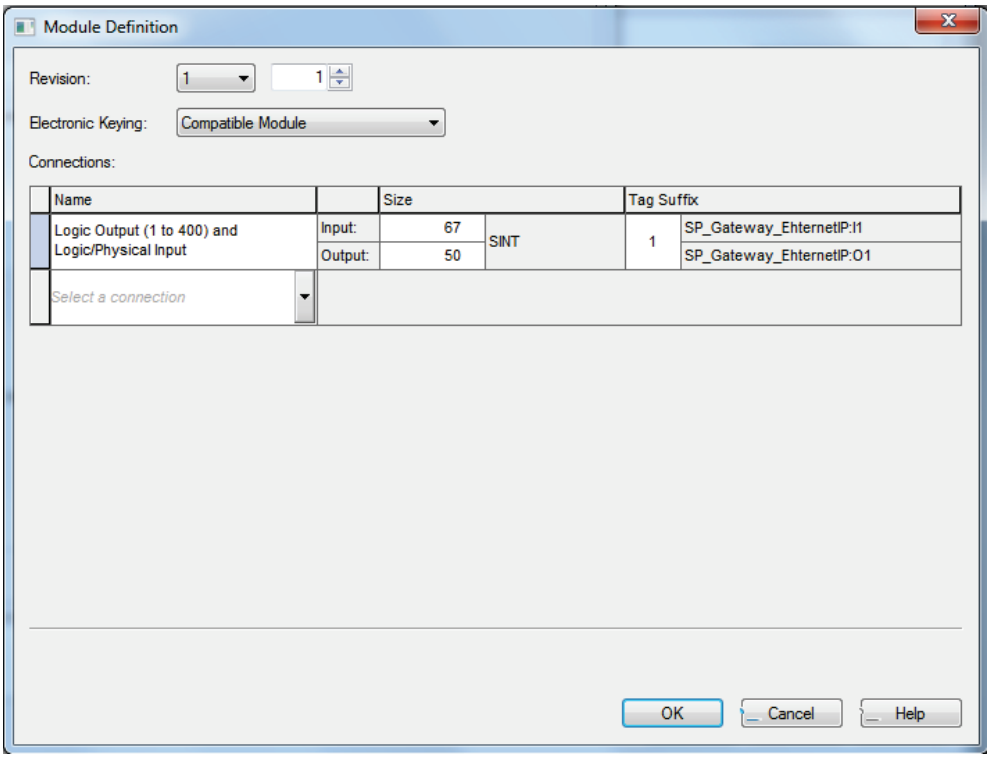


Illustration 19: Basic connection chosen in Logix Designer

This connection contains up to 50 bytes for data transfer from the PLC to the SP-COP2-ENI (entitled as assembly instance 37). The connection also contains up to 67 bytes for data transfer from the SP-COP2-ENI to the PLC (entitled as assembly instance 57). The following table gives an overview of these data bytes.

Table 30: Data of Class 1 Connection "Logic Output (1 to 400) and Logic/Physical Input"

Instance	Byte	Access	Data type	Meaning	Size	Data range
37	0 to 49	set, get	BYTE[50]	Output Bits, configured by **Input data block 1 to 5** in samosPLAN 5+ (*more [ch. 8.5.2.1, p. 97]*)	1 to 50 Byte	0 to 0xff
57	0 to 49	get	BYTE[50]	Logic Input Bits, configured by **Output data set 1** in samosPLAN5+ (*more [ch. 8.5.3.1, p. 98]*)	1 to 50 Byte	0 to 0xff
	50 to 65	get	BYTE[16]	Physical Input Bits (Instance 401 to 528 of attribute 3 class 8, not currently shown in samosPLAN5+) (*more [ch. 8.5.3.1, p. 98]*)	1 to 16 Byte	0 to 0xff
	66	get	BYTE	Bit 7: Input Status Bit 6: Output Status (not currently shown in samosPLAN5+)	1 Byte	0x00, 0x40, 0x80, 0xc0

Further connections supported by SP-COP2-ENI, are listed in the following table. For details of these assembly instances, please see table *Overview of Assembly data bytes in SP-COP2-ENI [ch. 8.5.1, p. 95]*.

Table 31: Class 1 Connections supported by SP-COP2-ENI

Connection name	Assembly for data from PLC to SP-COP (O→T)	Assembly for data from SP-COP to PLC (T→O)
Logic Output (1 to 400) and Logic/Physical Input	37	57
Logic Output (1 to 400) and Logic/Status/System Mode Assembly	37	167
Logic Output (81 to 400) and Logic/Physical Input	138	57
Logic Output (81 to 400) and Logic/Status/System Mode Assembly	138	167
Logic Output (161 to 400) and Logic/Physical Input	139	57
Logic Output (161 to 400) and Logic/Status/System Mode Assembly	139	167
Logic Output (241 to 400) and Logic/Physical Input	140	57
Logic Output (241 to 400) and Logic/Status/System Mode Assembly	140	167
Logic Output (321 to 400) and Logic/Physical Input	141	57
Logic Output (321 to 400) and Logic/Status/System Mode Assembly	141	167
(Listen Only) Logic/Physical Input	199	57

Connection name	Assembly for data from PLC to SP-COP (O→T)	Assembly for data from SP-COP to PLC (T→O)
(Listen Only) Logic/Status/System Mode Assembly	199	167
(Input Only) Logic/Physical Input	198	57
(Input Only) Logic/Status/System Mode Assembly	198	167

The Connection Point 199 (= 0xc7) is provided for **Listen Only** and the Connection Point 198 (= 0xc6) is provided for **Input Only**. Both are provided to have zero data size assembly, which means the PLC does not provide data to the SP-COP2-ENI.

If the PLC needs only process data from SP-COP, **Input Only** mode is recommended to the user.

8.3.2 Basic configuration of SP-COP2-ENI

The SP-EN-IP integrated gateway (EtherNet/IP Gateway) is part of the SP-COP2-ENI device and is activated by the gateway configuration box in the samosPLAN5+ software:

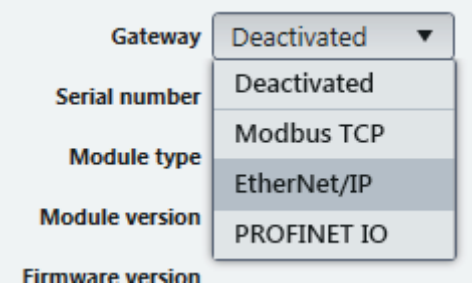


Illustration 20: Activation of EtherNet/IP in samosPLAN5+

The selection of the SP-EN-IP gateway is possible for SP-COP2-ENI modules version D-xx or later:

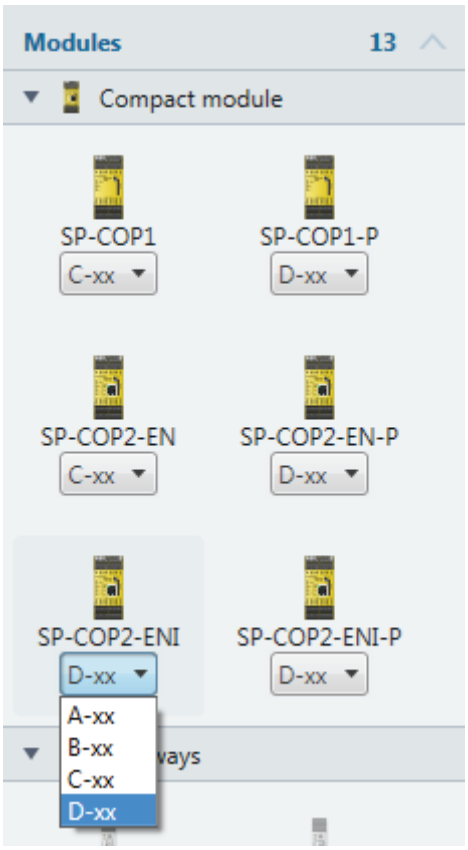


Illustration 21: Selection of module type for EtherNet/IP in samosPLAN5+

The IPv4 data of SP-COP2-ENI can be set in samosPLAN 5+ to match the settings of the PLC. The device must not be in **Run** mode for transferring IPv4 data.

Illustration 22: Setting IPv4 data of device via samosPLAN 5+

Stop the device if needed:

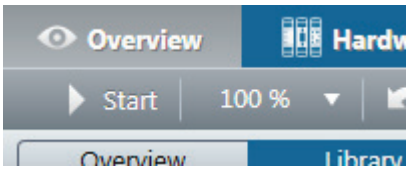


Illustration 23: "Start" button indicating device is not in "Run" mode

8.3.3 Configuring the data to PLC

The data sent to the PLC, meaningfrom target to originator (T→O), can be configured in the gateway configuration tab **samosPRO COMPACT → SP-EN-IP[0]** of samosPLAN5+. By default, the first 3 bytes contains data for input terminals I1 to I16 (and IQ1 to IQ4 if configured as inputs). Byte 4 contains data of output terminals Q1 to Q4 (and IQ1 to IQ4 if configured as outputs).

Bytes 12 to 23 contain data for input terminals I1 to I8 of extension modules SDI(O) or DIO. Bytes 24 to 35 contain data for output terminals Q1 to Q4 of extension modules SDIO or DIO Bytes 4 to 11 contain data from the logic editor and are called **Direct Outs** e.g. **Direct-out 1**.

This default configuration can be modified by Drag'n'Drop in the left "fly out" menu of the gateway configuration tabs as shown in the following figure:

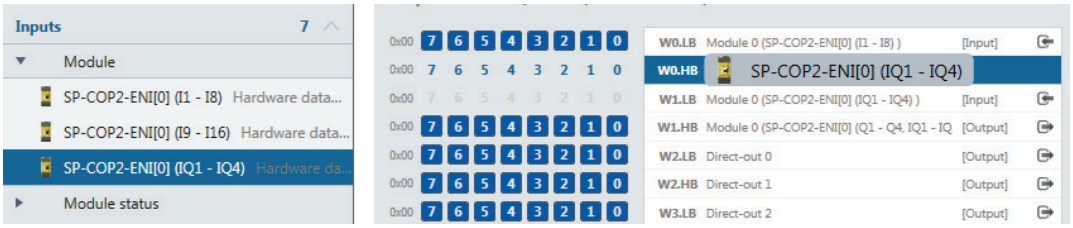


Illustration 24: Adding bytes into the gateway process data image (T→O) via Drag'n' Drop in samosPLAN5+

Tag names for all bytes can also be added or modified in samosPLAN5+ to use in the logic editor of samosPLAN5+ . Custom names improve program legibility and debugging. Tag names can be configured in the **Parameters** section of the "fly out" window on the right side.

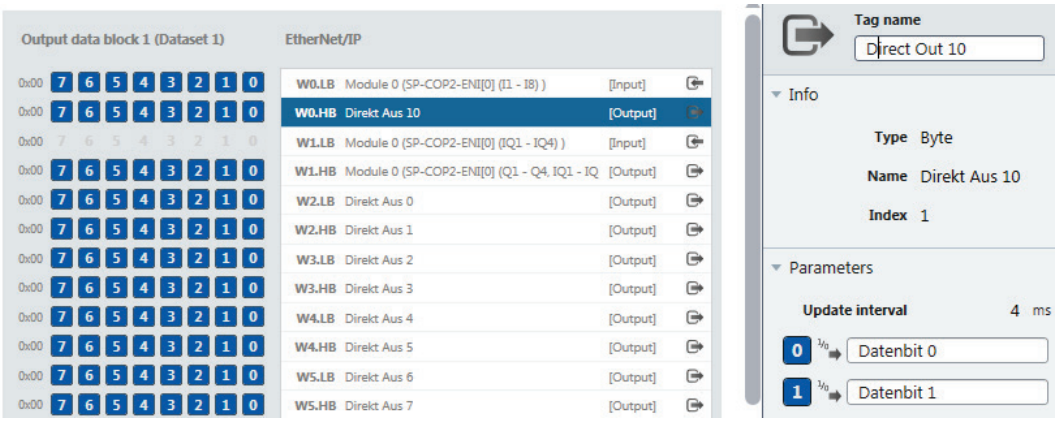


Illustration 25: Adding new data bytes (T→O) for use in the logic editor by configuring the tag names

8.3.4 Configuring the usage of data from PLC

Data sent from the PLC, meaning from originator to target (O→T), can be named in the gateway configuration tab **SP-EN-IP[0] → samos®PRO COMPACT** of samosPLAN5+. The first four bytes are pre-allocated with **Direct-in 0** to **Direct-in 3**. Each bit is pre-allocated with the name **Data bit 0** to **Data bit 7** and can be used in the logic editor of samosPLAN5+ as a non-safe input element e.g. **Restart button** or **Lamp**.

If further gateway data input elements are needed, they can be added by configuring additional tag names.

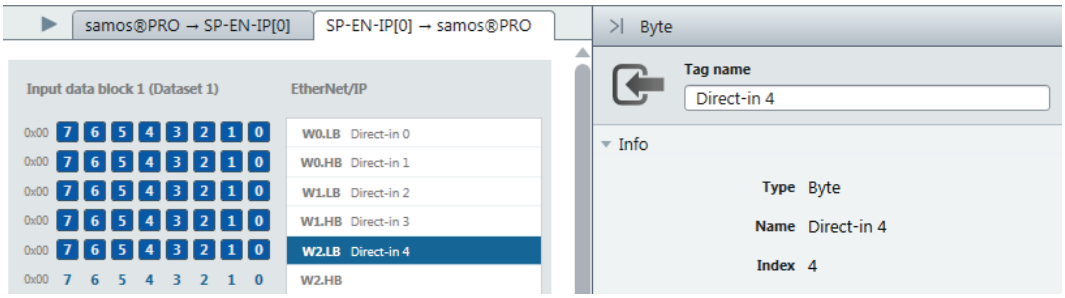


Illustration 26: Adding new data byte (O→T) for use in the logic editor by configuring the tag name.

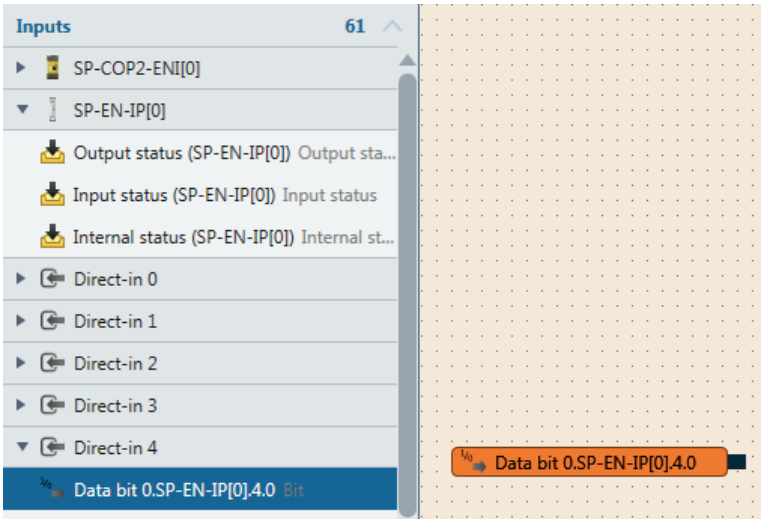


Illustration 27: Adding gateway data elements via Drag'n'Drop in the logic editor of samosPLAN5+

8.4 Supported CIP Objects

8.4.1 Identity Object

The Identity Object is required for all EtherNet/IP based products. Instance 1, attribute 1 represents the Vendor ID. For Wieland Electric GmbH, the value 314 is registered by ODVA.

Instance 1, attribute 2 represents the device type. The open type code 0x07 means **General Purpose Discrete I/O Device**.

Instance 1, attribute 3 represents the product code. Its type is UINT, means 2 bytes, and the decimal value 1320 is derived from device order number R1.190.1320.0.

Instance 1, attribute 4 represents the revision, including SP-COP2-ENI major and minor firmware version which can be found as **Diagnosis version** in the **Hardware** view and as **Firmware version** in the **Gateway** view in samosPLAN5+ software.

Instance 1, attribute 5 represents the current status of the entire device. The data range is listed in table *Device Status values of SP-COP2-ENI in class 1, instance 1, attribute 5 [ch. 8.4.1, p. 79]*.

Instance 1, attribute 6 represents the device's serial number, which can be checked in samosPLAN5+ under hardware configuration Instance 1, attribute 7 represents the product's name which is **SP-COP2-ENI**.

Table 32: Overview of Identity Class (0x01) supported by SP-COP2-ENI

Class	Instance	Attribute	Access	Data type	Meaning	Data range
1	0 = Class	1	get	UINT	Revision	1
1	0 = Class	2	get	UINT	Maximum Instance	1
1	0 = Class	3	get	UINT	Number of Instances	1
1	0 = Class	6	get	UINT	Maximum Class Attribute ID	7
1	0 = Class	7	get	UINT	Maximum Instance Attribute ID	7
1	1	1	get	UINT	Vendor ID	314 = 0x13a
1	1	2	get	UINT	Device Type	0x07
1	1	3	get	UINT	*Product Code [ch. 8.4.8.3, p. 92]*	1320
1	1	4	get	USINT[2]	Revision, Software Version 'Left' Byte is the major part and is transferred first	{1, 1} to {99, 99}
1	1	5	get	WORD	Device Status	see next table
1	1	6	get	UDINT	Serial Number	16010001 to 99539999
1	1	7	get	SHORT_ STRING	Product Name	SP-COP2-ENI

Table 33: Device Status values of SP-COP2-ENI in class 1, instance 1, attribute 5

Status value	Meaning	System mode values
0b0000 xxxx xxxx 0x01	Device has owner	4 = Idle = Stop 5 = Run = Executing 7 = Critical Fault 21 = Force mode
0b0000 xxxx 0000 010x	Device is configured	4 = Idle = Stop 5 = Run = Executing 7 = Critical Fault 21 = Force mode

Status value	Meaning	System mode values
0b0000 0001 0000 0x0x	Minor Recoverable Fault	4 = Idle = Stop 5 = Run = Executing 21 = Force mode
0b0000 0010 0000 0x0x	Minor Unrecoverable Fault	4 = Idle = Stop 5 = Run = Executing 21 = Force mode
0b0000 0100 0000 0x0x	Major Recoverable Fault	1 = Init 2 = Configuration required 3 = Configuring
0b0000 1000 0000 0x0x	Major Unrecoverable Fault	7 = Critical Fault

8.4.2 Assembly Object

All data of class 1 connections are also provided by assembly object. The following table gives an overview of assembly object.

Further information:

- Table *Overview of Assembly data bytes in SP-COP2-ENI [ch. 8.5.1, p. 95]*
- Figure Bild 13: Data flow with usage of Assembly Instances provides by SP-COP2-ENI (Linkziel)
  (Illustrates the data flow from the PLC to SP_COP2-ENI and back from the perspective of several assemblies.)

Table 34: Overview of Assembly Class (0x04) supported by SP-COP2-ENI

Class	Instance	Attribute	Access	Data type	Meaning	Data range
4	0 = Class	1	get	UINT	Class Revision	2
4	0 = Class	2	get	UINT	Maximum Instance	167
4	0 = Class	3	get	UINT	Number of Instances	7
4	0 = Class	6	get	UINT	Maximum Class Attribute ID	7
4	0 = Class	7	get	UINT	Maximum Instance Attribute ID	4
4	37	1	get	UINT	Number Of Members	0
4	37	3	get, set	BYTE[50]	*Logic Output Bits [ch. 8.5.2, p. 97]* (Instance 1 to 400 of Class 9)	see[1)]
4	37	4	get	UINT	Number of data bytes	50
4	57	1	get	UINT	Number Of Members	0
4	57	3	get	BYTE[67]	Input Bits (Instance 1 to 528 of Class 8)	see[1)]
4	57	4	get	UINT	Number of data bytes	67
4	138	1	get	UINT	Number Of Members	0
4	138	3	get, set	BYTE[40]	*Logic Output Bits [ch. 8.5.2, p. 97]* (Instance 81 to 400 of Class 9)	see[1)]
4	138	4	get	UINT	Number of data bytes	40
4	139	1	get	UINT	Number Of Members	0
4	139	3	get, set	BYTE[30]	*Logic Output Bits [ch. 8.5.2, p. 97]* (Instance 161 to 400 of Class 9)	see[1)]
4	139	4	get	UINT	Number of data bytes	30
4	140	1	get	UINT	Number Of Members	0
4	140	3	get, set	BYTE[20]	*Logic Output Bits [ch. 8.5.2, p. 97]* (Instance 241 to 400 of Class 9)	see[1)]
4	140	4	get	UINT	Number of data bytes	20
4	141	1	get	UINT	Number Of Members	0
4	141	3	get, set	BYTE[10]	*Logic Output Bits [ch. 8.5.2, p. 97]* (Instance 321 to 400 of Class 9)	see[1)]
4	141	4	get	UINT	Number of data bytes	10
4	167	1	get	UINT	Number Of Members	0
4	167	3	get	BYTE[112]	Logic Input Bits, Mode and Status bytes (*more [ch. 8.5.3.2, p. 98]*)	see[1)]
4	167	4	get	UINT	Number of data bytes	112
[1)]See: Table *Overview of Assembly data bytes in SP-COP2-ENI [ch. 8.5.1, p. 95]*						

8.4.3 Discrete Input Point Object

The Discrete Input Point Objects are part of the device profile **General Purpose Discrete I/O Device**.

When a terminal input of a particular instance from 401 to 528 has a failure and SP-COP2-ENI is in **Run** mode, the value of the instance's attribute 4 is 1. In all other cases, the value is 0.

Table 35: Overview of Discrete Input Point Object (0x08) supported by SP-COP2-ENI

Class	Instance	Attribute	Access	Data type	Meaning	Data range
8	0 = Class	1	get	UINT	Class Revision	2
8	0 = Class	2	get	UINT	Maximum Instance	584
8	0 = Class	3	get	UINT	Number of Instances	400 + 128 + 56 logic + input + output
8	0 = Class	6	get	UINT	Maximum Class Attribute ID	7
8	0 = Class	7	get	UINT	Maximum Instance Attribute ID	4
8	1 to 400 and 529 to 584	1	get	USINT	Number of Attributes	3
8	401 to 528	1	get	USINT	Number of Attributes	4
8	1 to 528	2	get	USINT[4]	Supported Attributes List	{1, 2, 3, 4}
8	529 to 584	2	get	USINT[3]	Supported Attributes List	{1, 2, 3}
8	1 to 400	3	get	BOOL	Value of Input Bit, configured by **Output data set 1** in samosPLAN5+, means data from SP-COP´s logic to PLC	0 = off, 1 = on
8	1 to 400	4	get	BOOL	Status of **Output data set 1**	0 = ok
8	401 to 416	3	get	BOOL	Value of terminal I1 to I16 of SP-COP2-ENI	0, 1
8	401 to 416	4	get	BOOL	Status of terminal I1 to I16 of SP-COP2-ENI	0, 1
8	417 to 420	3	get	BOOL	Value of terminal IQ1 to IQ4 of SP-COP2-ENI if configured as input	0, 1
8	417 to 420	4	get	BOOL	Status of terminal IQ1 to IQ4 of SP-COP2-ENI if configured as input	0, 1
8	421 to 430	3	get	BOOL	Reserved	0
8	431	3	get	BOOL	Value of B1	voltage is... 0 = out of range 1 = in range
8	432	3	get	BOOL	Value of B2	voltage is... 0 = out of range 1 = in range
8	421 to 432	4	get	BOOL	Reserved	0

Class	Instance	Attribute	Access	Data type	Meaning	Data range
8	425 + 8 x n to 432 + 8 x n	3	get	BOOL	Value of terminal I1 to I8 of SP-SDI[n] / SP-SDIO[n], with n = 1 to 12	0, 1
8	425 + 8 x n to 432 + 8 x n = 528	4	get	BOOL	State of terminal I1 to I8 of SP-SDI[n] / SP-SDIO[n], with n = 1 to 12	0, 1
8	529 to 532	3	get	BOOL	Value of terminal Q1 to Q4 of SP-COP2-ENI	0, 1
8	533 to 536	3	get	BOOL	Value of terminal IQ1 to IQ4 of SP-COP2-ENI if configured as output	0, 1
8	533 + 4 x n to 536 + 4 x n = 584	3	get	BOOL	Value of terminal Q1 to Q4 of SP-SDIO[n], with n = 1 to 12	0, 1

8.4.4 Discrete Output Point Object

The Discrete Output Point Objects are part of the device profile **General Purpose Discrete I/O Device**.

The SP-COP does not allow direct setting of safety output terminals. Instead, PLCs can set up to 400 data bits. These are accessible as bits in the **Input data blocks 1 to 5** in samosPLAN5+. The simplest way to control output terminals from a PLC is to connect the respective gateway bit to an output in the logic editor of samosPLAN5+. The following figure shows an example:

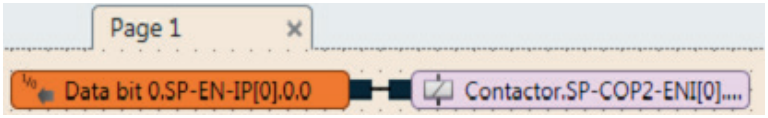


Illustration 28: Direct connection of a gateway input bit to an output terminal of SP-COP2-ENI

ATTENTION

Check your application thoroughly for correctness!
Because the samosPLAN5+ only checks for logic-internal connection errors, you have to check the following aspects systematically yourself:

- Does your application correspond to the results from the risk analysis and the avoidance strategy?
- Have all of the applicable standards and guidelines been complied with?
  If not, you are placing the machine's operator in danger.

Note, the default value of output terminal is **off**, which means value “0”. This value is always used when the SP-COP is not in **Run** mode or when the output is not configured via the logic editor in samosPLAN5+.

The default value of gateway output bits can be configured by attributes 5 and 6.

Instance attribute 5 controls whether the gateway data bit is set or not set in the case that the connection is lost between the PLC and the SP-COP. The set value is controlled by instance attribute 6.

A write request to attribute 3 of instances 1 to 400 is rejected if Assembly *Instance 37 [ch. 8.5.2.1, p. 97]* is linked to an active I/O connection.

A write request to attribute 3 of instances 81 to 400 is rejected if Assembly *Instance 138 [ch. 8.5.2, p. 97]* is linked to an active I/O connection.

A write request to attribute 3 of instances 161 to 400 is rejected if Assembly *Instance 139 [ch. 8.5.2, p. 97]* is linked to an active I/O connection.

A write request to attribute 3 of instances 241 to 400 is rejected if Assembly *Instance 140 [ch. 8.5.2, p. 97]* is linked to an active I/O connection.

A write request to attribute 3 of instances 321 to 400 is rejected if Assembly *Instance 141 [ch. 8.5.2, p. 97]* is linked to an active I/O connection.

Table 36: Overview of Discrete Output Point Object (0x09) supported by SP-COP2-ENI

Class	Instance	Attribute	Access	Data type	Meaning	Data range
9	0 = Class	1	get	UINT	Class Revision	1
9	0 = Class	2	get	UINT	Maximum Instance	400
9	0 = Class	3	get	UINT	Number of Instances	400
9	0 = Class	6	get	UINT	Maximum Class Attribute ID	7
9	0 = Class	7	get	UINT	Maximum Instance Attribute ID	6
9	1 to 400	1	get	USINT	Number of attributes	5
9	1 to 400	2	get	USINT[5]	Supported Attributes List	{1, 2, 3, 5, 6}
9	1 to 400	3	set, get	BOOL	Value of Logic Output Bit, configured by **Input data block 1 to 5** in samosPLAN5+, means data from PLC to SP-COP´s logic	0 = off, 1 = on
9	1 to 400	5	set, get	BOOL	Fault action (Value set at connection loss to PLC)	0 = Fault value, 1 = last state
9	1 to 400	6	set, get	BOOL	Fault value	0 = off, 1 = on

8.4.5 Discrete Input Group Object

The Discrete Input Group Objects are part of the device profile **General Purpose Discrete I/O Device**.

The object of class 29 plays the role of the alarm bit. It collects the process alarms of all physical inputs of SP-COP2-ENI and SDI(O) extension modules into one bit. When at least one physical input has a failure and SP-COP2-ENI is in **Run** mode, the value of instance 1 attribute 5 is 1. In all other cases, the value is 0.

Table 37: Overview of Discrete Input Group Object (0x1D) supported by SP-COP2-ENI

Class	Instance	Attribute	Access	Data type	Meaning	Data range
29	0 = Class	1	get	UINT	Class Revision	1
29	0 = Class	2	get	UINT	Maximum Instance	1
29	0 = Class	3	get	UINT	Number of Instances	1
29	0 = Class	6	get	UINT	Maximum Class Attribute ID	7
29	0 = Class	7	get	UINT	Maximum Instance Attribute ID	5
29	1	1	get	USINT	Number of attributes	5
29	1	2	get	USINT[5]	Supported Attributes List	{1, 2, 3, 4, 5}
29	1	5	get	BOOL	Group status of all physical inputs (state of instances 401 to 420 of class 8)	0 = no failure, 1 = failure

8.4.6 Discrete Output Group Object

The Discrete Output Group Objects are part of the device profile **General Purpose Discrete I/O Device**.

The object of class 30 plays the role of the alarm bit. It collects the process alarms of all physical outputs of SP-COP2-ENI and SDIO extension modules into one bit. When at least one physical output has a failure and SP-COP2-ENI is in **Run** mode, the value of instance 1 attribute 5 is 1. If SP-COP2-ENI is in **Critical fault** mode, this attribute is also 1. In all other cases, the value is 0.

Table 38: Overview of Discrete Output Group Object (0x1E) supported by SP-COP2-ENI

Class	Instance	Attribute	Access	Data type	Meaning	Data range
30	0 = Class	1	get	UINT	Class Revision	1
30	0 = Class	2	get	UINT	Maximum Instance	1
30	0 = Class	3	get	UINT	Number of Instances	1
30	0 = Class	6	get	UINT	Maximum Class Attribute ID	7
30	0 = Class	7	get	UINT	Maximum Instance Attribute ID	6
30	1	1	get	USINT	Number of attributes	6
30	1	2	get	USINT[6]	Supported Attributes List	{1, 2, 3, 4, 5, 6}
30	1	3	get	USINT	Number of bound instances	56
30	1	4	get	UINT[56]	Bound Instances	{1, ..., 56}
30	1	5	get	BOOL	Group status of all physical outputs (state of instances 529 to 584 of class 8)	0 = no failure, 1 = failure

8.4.7 PCCC Object

PCCC (spoken "p c cube") is used by several PLCs from Rockwell Automation / Allen Bradley that are still used in the market. It was designed before CIP and EtherNet/IP were defined. PCCC telegrams are either:

a) Encapsulated in CIP packets (e.g. via EtherNet/IP)

b) Encapsulating CIP packets.

SP-COP2-ENI supports the encapsulation of PCCC data inside CIP packages, according to item b) noted above. For this, class ID 0x67 = 103 is defined.

PCCC commands listed in the following table are supported by the SP-COP2-ENI.

All PCCC related data with a size of 16 bits (word) having little endian format. This means the least significant byte comes first.

Table 39: PCCC commands supported by SP-COP2-ENI

Type	CMD	FNC	Meaning	Command supported by
PLC-5	0x0f	0x00	*Word Range Write [ch. 8.4.7.2, p. 87]*	PLC-3, PLC-5, PLC-5/250
PLC-5	0x0f	0x01	*Word Range Read [ch. 8.4.7.3, p. 87]*	PLC-3, PLC-5, PLC-5/250
PLC-5	0x0f	0x67	*Typed Write [ch. 8.4.7.4, p. 87]*	SLC 5/03, SLC 5/04, PLC 5, PLC-5/250, PLC-5/VME
PLC-5	0x0f	0x68	*Typed Read [ch. 8.4.7.5, p. 88]*	SLC 5/03, SLC 5/04, PLC 5, PLC-5/250, PLC-5/VME
SLC	0x0f	0xa1	*Protected Typed Logical Read with 2 Address Fields [ch. 8.4.7.6, p. 90]*	
SLC	0x0f	0xa2	*Protected Typed Logical Read with 3 Address Fields [ch. 8.4.7.8, p. 90]*	MicroLogix-1000, SLC 500, SLC 5/03, SLC 5/04, PLC 5
SLC	0x0f	0xa9	*Protected Typed Logical Write with 2 Address Fields [ch. 8.4.7.7, p. 90]*	
SLC	0x0f	0xaa	*Protected Typed Logical Write with 3 Address Fields [ch. 8.4.7.9, p. 91]*	MicroLogix-1000, SLC 500, SLC 5/03, SLC 5/04

8.4.7.1 PCCC Telegram Structure

Each request telegram contains 7+5 header bytes.

Table 40: PCCC request header

Name	Data type	Meaning	Size	Data range
Length	USINT	Header size	1 Byte	7
Vendor	UINT	Vendor ID of requester	2 Byte	
S/N	UDINT	Serial number of requester	4 Byte	0 to 2^{32}-1
CMD	USINT	Command	1 Byte	0x0f
STS	USINT	Status	1 Byte	0
TNSW	UINT	Transport sequence number	2 Byte	1 to 65535
FNC	USINT	Function Code	1 Byte	0x67, 0x68, 0xa2, 0xaa

Each reply telegram contains 7+4 header bytes, or 7+4+1 header bytes if the status byte is 0xf0.

Table 41: PCCC reply header

Name	Data type	Meaning	Size	Data range
Length	USINT	Header size	1 Byte	7
Vendor	UINT	Vendor ID of requester	2 Byte	
S/N	UDINT	Serial number of requester	4 Byte	0 to 2^{32}-1

Name	Data type	Meaning	Size	Data range
CMD	USINT	Command of requester plus Bit 6 set	1 Byte	0x4f
STS	USINT	Status	1 Byte	0x00, 0x10, 0xf0
TNSW	UINT	Transport sequence number	2 Byte	1 to 65535
EXT STS	USINT	Extended Status, only present if STS = 0xf0	0 to 1 Byte	

8.4.7.2 Word Range Write

SP-COP2-ENI supports PLC-5 Word Range Write according to following table:

Table 42: Data structure of PLC-5 Word Range Write

Name	Data type	Meaning	Data range
Packet offset	UINT	Offset in number of elements	
Total Transaction	UINT	number of elements in transaction	
Address	BYTE[m]	PLC-5 system address, m >= 2	
Payload	UINT[n]	2 x n = Data byte count	0 to 65535

The reply of SP-COP2-ENI does not contain data, only state.

8.4.7.3 Word Range Read

SP-COP2-ENI supports PLC-5 Word Range Read according to following table:

Table 43: Request data structure of PLC-5 Word Range Read

Name	Data type	Meaning	Data range
Packet offset	UINT	Offset in number of elements	
Total transaction	UINT	Number of elements in transaction	0 to assembly size dependent value
Address	BYTE[m]	PLC-5 system address, m >= 2	'0' to ':', 'A' to 'Z', 'a' to 'z'
Size	UINT	Number of elements to return	

Table 44: Reply for SP-COP2-ENI of PLC-5 Word Range Read

Name	Data type	Meaning	Data range
Payload	UINT[n]	2•n = Data byte count (up to 244 bytes)	0 to 65535

8.4.7.4 Typed Write

SP-COP2-ENI supports PLC-5 Typed Write according following table:

Table 45: Data structure of PLC-5 Typed Write

Name	Data type	Meaning	Data range
Packet offset	UINT	Offset in number of elements	
Total transaction	UINT	Number of elements in transaction	
Address	BYTE[m]	PLC-5 system address, m >= 2	see next table
Type ID	BYTE[n]	Type and size of data, n >= 1	

The reply of SP-COP2-ENI does not contain data, only state, see table *PCCC reply header [ch. 8.4.7.1, p. 86]*. The data format UINT is the same as for Word Range Write.

Table 46: Address scheme for PLC-5 Typed Write

Address	Data type	Element count	Meaning	Data range
$N37:x	UINT[n]	n	Output Assembly of General Purpose Discrete I/O Device profile, x = 0 to 24, n = 25 - x	0 to 65535
$N138:x	UINT[n]	n	Output Assembly of Logic Output, configured by **Input data block 2** in samosPLAN5+, x = 0 to 19, n = 20 - x	0 to 65535
$N139:x	UINT[n]	n	Output Assembly of Logic Output, configured by **Input data block 3** in samosPLAN5+, x = 0 to 14, n = 15 - x	0 to 65535
$N140:x	UINT[n]	n	Output Assembly of Logic Output, configured by **Input data block 4** in samosPLAN5+, x = 0 to 9, n = 10 - x	0 to 65535
$N141:x	UINT[n]	n	Output Assembly of Logic Output, configured by **Input data block 5** in samosPLAN5+, x = 0 to 4, n = 5 - x	0 to 65535
The valid ranges of the element counts of the specific address blocks in this table are related to the assembly sizes noted here: Table *Overview of Assembly data bytes in SP-COP2-ENI [ch. 8.5.1, p. 95]*				

8.4.7.5 Typed Read

SP-COP2-ENI supports PLC-5 Typed Read according the following table:

Table 47: Request data structure of PLC-5 Typed Read

Name	Data type	Meaning	Data range
Packet offset	UINT	Offset in number of elements	
Total transaction	UINT	Number of elements in transaction	0 to assembly size dependent value
Address	BYTE[m]	PLC-5 system address, m >= 2	'0' to ':', 'A' to 'Z', 'a' to 'z'
Size	UINT	Number of elements to return	

The reply of SP-COP2-ENI is listed in the following table. The first byte of Type ID is 0x9a = 0b1001 1010 which means the data type value is in the following bytes and data size is in the next byte. The fourth byte of Type ID is 0x42 = 0b0100 0010 which means integer data type of size 2.

Table 48: Reply for SP-COP2-ENI for data structure of PLC-5 Typed Read

Name	Data type	Meaning	Data range
Type ID	BYTE	Type and size of data	Bit 0 to 3: 10 = size in after next byte Bit 4 to 7: 9 = type in next byte
Type ID	BYTE	Data type	9 = array of similar elements
Type ID	BYTE	Number of following Bytes	1 to n+1
Type ID	BYTE	Type and size of data	Bit 0 to 3: 2 = UINT Bit 4 to 7: 4 = integer
Payload	UINT[n]	2•n = data byte count	0 to 65535

With Typed Read, command data from all Assembly Instances can be acquired.

Contrary to native EtherNet/IP assembly instance addressing, the PLC-5 system address contains element offset, which can be used by the requester.

SP-COP2-ENI supports array of UINT as PCCC data types. Due to odd size of assembly instance 57, the firmware in SP-COP2-ENI allocates one additional byte for providing an even number of bytes for the PLC-5 Typed Read.

The address scheme SP-COP2-ENI supports for PLC-5 Typed Read is given by the following table:

Table 49: Address scheme for PLC-5 Typed Read

Address	Data type	Element count	Meaning	Data range
$N57:x	UINT[n]	n	Input Assembly of General Purpose Discrete I/O Device profile, x = 0 to 33, n = 34 - x	Element 1 to 33: 0 to 65535 Element 34 Bit 0 to 7 (LSB): 0x00, 0x40, 0x80, 0xc0 Element 34 Bit 8 to 15 (MSB): 0
$N167:x	UINT[n]	n	Input Assembly of:	
			Logic Inputs Bits (n = 1-x to 25-x, x = 0 to 24)	0 to 65535
			System Status and System Mode (n = 26-x, x = 0 to 25)	Bit 0 to 7 (LSB): System Mode (1, 2, 3, 4, 5, 7, 21) Bit 8 to 15 (MSB): System Status (0x00, 0x40, 0x80, 0xc0)
			SP-COP Status Bytes (n = 27-x to 56-x, x = 26 to 55)	0 to 65535

Example: “$N57:10” and “Total Transaction = 24” addresses elements 11 to 34, which are byte 20 to 66 of Assembly instance 57.

Note: Byte 67 will also be sent which is not defined in the Assembly instance 57.

Note: Position of Word data containing System Status and System Mode depends on requested data count “x”.

8.4.7.6 Protected Typed Logical Read with 2 Address Fields

SP-COP2-ENI supports SLC Protected Typed Logical Read according to following table:

Table 50: Request data structure of SLC Protected Typed Logical Read with 2 Address Fields

Name	Data type	Meaning	Data range
Byte size	USINT	Number of data bytes to be read	Assembly Instance 37: 0 to 50 Assembly Instance 57: 0 to 67 Assembly Instance 167: 0 to 112
File number	USINT	Assembly Instance ID	37, 57, 167
File type	USINT	Data type	0x89 = Integer data
Element number	USINT	Offset = ID of first element in reply	Assembly Instance 37: 0 to 24 – size/2 Assembly Instance 57: 0 to 33 – size/2 Assembly Instance 167: 0 to 55 – size/2

Table 51: Reply for SP-COP2-ENI of SLC Protected Typed Logical Read with 2 Address Fields

Name	Data type	Meaning	Data range
Payload	UINT[n]	2•n = data byte count	0 to 65535

8.4.7.7 Protected Typed Logical Write with 2 Address Fields

SP-COP2-ENI supports SLC Protected Typed Logical Write according to following table:

Support of Assembly Instances 138 to 141 is not needed. Offset, means first byte, is set by **Element Number** instead.

Table 52: Request data structure of SLC Protected Typed Logical Write with 2 Address Fields

Name	Data type	Meaning	Data range
Byte size	USINT	Number of data bytes to be written	0 to 50
File number	USINT	Assembly Instance ID	37
File type	USINT	Data type	0x89 = Integer data
Element number	USINT	Offset = ID of first element to be written	0 to 24 – size/2
Payload	UINT[n]	n = size/2	0 to 65535

8.4.7.8 Protected Typed Logical Read with 3 Address Fields

SP-COP2-ENI supports SLC Protected Typed Logical Read according the following table:

Table 53: Request data structure of SLC Protected Typed Logical Read with 3 Address Fields

Name	Data type	Meaning	Data range
size	USINT	Number of data bytes to be read	Assembly Instance 37: 0 to 50 Assembly Instance 57: 0 to 67 Assembly Instance 167: 0 to 112
file number	USINT	Assembly Instance ID	37, 57, 167

Name	Data type	Meaning	Data range
file type	USINT	Data type	0x89 = Integer data
element number	USINT	Offset = ID of first element in reply	Assembly Instance 37: 0 to 24 – size/2 Assembly Instance 57: 0 to 33 – size/2 Assembly Instance 167: 0 to 55 – size/2
sub-element	USINT	don't care	0 to 254 (for 1 byte size)

8.4.7.9 Protected Typed Logical Write with 3 Address Fields

SP-COP2-ENI supports SLC Protected Typed Logical Write according to following table:

Table 54: Request data structure of SLC Protected Typed Logical Write with 3 Address Fields

Name	Data type	Meaning	Data range
size	USINT	Number of data bytes to be written	0 to 50
file number	USINT	Assembly Instance ID	37
file type	USINT	Data type	0x89 = Integer data
element number	USINT	Offset = ID of first element in reply	0 to 25 – size/2
sub-element	USINT	don't care	0 to 254 (for 1 byte size)
Payload	UINT[n]	n = size/2	0 to 65535

8.4.8 Vendor Object

The Vendor Object with class ID = 0x78 provides CRC, status and diagnostic data which are not covered by device profile **General Purpose Discrete I/O Device**. Furthermore it supplies an interface to input and output data in a compressed and therefore network traffic saving format.

Note that several instances have different attribute types and numbers. Packing several data together into this vendor object class is made for legacy reasons.

8.4.8.1 Instance 1

The instance 1, attributes 1 to 50 provide Input Bytes, configured by **Output data set** 1 in samosPLAN5+, which means data from SP-COP´s logic to the PLC.

8.4.8.2 Instance 2

The instance 2, attribute 1 provides the CRC of the active project file, created by samosPLAN5+. The instance 2, attributes 2 to 8 are reserved for future use.

8.4.8.3 Instance 3

The instance 3, attributes 1 to 60, provide status bytes. The meaning of each bit is listed in Table *Status bytes of SP-COP [ch. 8.4.8.7, p. 93]*. These data are similar to **Data set 3** as stated elsewhere in this document.

A Value = 1 for bits in instance 3, attributes 1 to 60 mean “OK” / “not used” / “reserved”. A Value = 0 means “failure” or “error” or “outside the limit”. “Don’t care” means, the value can be 0 or 1.

SP-COP2 means the SP-COP2-ENI controller module. SP-X[13] means the first and SP-X[14] means the second extension gateway module. SP-X[n] with n = 1 to 12 means the 1st to 12th safe SP-SDI or SP-SDIO or non safe SP-DIO extension module.

8.4.8.4 Instance 4

The instance 4, attributes 1 to 60 are reserved for future use. Values are zero and changes are reserved.

8.4.8.5 Instance 5

The attribute 1 of instance 5 provides the system state / mode of the SP-COP. The values are listed in the following table:

Table 55: System states / modes of SP-COP2-ENI

System state / mode	Value
Power Up	0
Init	1
Configuration required	2
Configuring	3
Idle = Stop	4
Run	5
Critical Fault	7
Force mode	21

8.4.8.6 Instance 6

The attribute 1 of instance 6 provides the error code of the last error in the SP-COP. A Value = 0 means no error has occurred. The attribute 2 provides the error code of the previous error and so on up to and including attribute 5.

8.4.8.7 Instance 7

The attributes 1 to 50 of instance 7 are the **Input data block 1 to 5** in samosPLAN5+, which means data from the PLC to the SP-COP´s logic.

The attributes 1 to 50 of instance 7 have same data as Assembly Instance 37 Byte 0 to 49.

Table 56: Overview of Vendor specific Object (Wieland Electric, 0x78) supported by SP-COP2-ENI

Class	Instance	Attribute	Access	Data type	Meaning	Data range
120	0 = Class	1	get	UINT	Class Revision	1
120	0 = Class	2	get	UINT	Maximum Instance	4
120	0 = Class	3	get	UINT	Number of Instances	4
120	0 = Class	5	get	UINT[3]	Optional service list	{2, 0x4c, 0x4d}
120	0 = Class	6	get	UINT	Maximum Class Attribute ID	7
120	0 = Class	7	get	UINT	Maximum Instance Attribute ID	60
120	1	n+1	get	USINT	Input Byte n, configured by **Output data set 1** in samosPLAN5+, means data from SP-COP´s logic to PLC, with n = 0 to 49	0 to 255
120	2	1	get	UDINT	Project file CRC (Data Set 2)	0 to 2^{32}-1
120	2	2 to 8	get	UDINT	Reserved (Data Set 2)	0
120	3	n+1	get	BYTE	SP-COP Status Byte n, with n = 0 to 59	0 to 255
120	4	n+1	get	BYTE	AP-COP2 auxiliary Byte n, with n = 0 to 59	0
120	5	1	get	USINT	SP-COP2-ENI System Mode (*more [ch. 8.4.8.5, p. 92]*)	1, 2, 3, 4, 5, 7, 21
120	6	n	get	UDINT	Error code in SP-COP, with n = 1 for error occurred last, n = 2 for error occurred before etc., with n = 1 to 5	0 to 2^{32}-1
120	6	1	set	UDINT	Clearing the error code queue of instance 6	0
120	7	n+1	set, get	BYTE	Output Byte n, configured by **Input data block 1 to 5** in samosPLAN5+, means data from PLC to SP-COP´s logic, with n = 0 to 49	0 to 255

Table 57: Status bytes of SP-COP

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	SP-COP2 B2 state	SP-COP2 collective error fast shut-off	SP-COP2 B1 state	SP-COP2 configura-tion state	SP-COP2 A1 state	SP-COP2 external module state	SP-COP2 internal module state	reserved
1	SP-COP2 output data state	SP-COP2 input data state	reserved	reserved	SP-COP2 IQ3+IQ4 current	SP-COP2 IQ1+IQ2 current	SP-COP2 Q3+Q4 current	SP-COP2 Q1+Q2 current
2	SP-COP2 I8 test pulse match	SP-COP2 I7 test pulse match	SP-COP2 I6 test pulse match	SP-COP2 I5 test pulse match	SP-COP2 I4 test pulse match	SP-COP2 I3 test pulse match	SP-COP2 I2 test pulse match	SP-COP2 I1 test pulse match

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
3	SP-COP2 I16 test pulse match	SP-COP2 I15 test pulse match	SP-COP2 I14 test pulse match	SP-COP2 I13 test pulse match	SP-COP2 I12 test pulse match	SP-COP2 I11 test pulse match	SP-COP2 I10 test pulse match	SP-COP2 I9 test pulse match
4	reserved	reserved	reserved	reserved	SP-COP2 IQ4 (input) test pulse match	SP-COP2 IQ3 (input) test pulse match	SP-COP2 IQ2 (input) test pulse match	SP-COP2 IQ1 (input) test pulse match
5	SP-COP2 I15/I16 two-channel state	SP-COP2 I13/I14 two-channel state	SP-COP2 I11/I12 two-channel state	SP-COP2 I9/I10 two-channel state	SP-COP2 I7/I8 two-channel state	SP-COP2 I5/I6 two-channel state	SP-COP2 I3/I4 two-channel state	SP-COP2 I1/I2 two-channel state
6	reserved	reserved	reserved	reserved	reserved	reserved	SP-COP2 IQ3/IQ4 two-channel state	SP-COP2 IQ1/IQ2 two-channel state
7	reserved	reserved	reserved	reserved	reserved	reserved	reserved	reserved
8	SP-COP2 Q4 Stuck at low	SP-COP2 Q4 Stuck at high	SP-COP2 Q3 Stuck at low	SP-COP2 Q3 Stuck at high	SP-COP2 Q2 Stuck at low	SP-COP2 Q2 Stuck at high	SP-COP2 Q1 Stuck at low	SP-COP2 Q1 Stuck at high
9	SP-COP2 IQ4 (Output) Stuck at low	SP-COP2 IQ4 (Output) Stuck at high	SP-COP2 IQ3 (Output) Stuck at low	SP-COP2 IQ3 (Output) Stuck at high	SP-COP2 IQ2 (Output) Stuck at low	SP-COP2 IQ2 (Output) Stuck at high	SP-COP2 IQ1 (Output) Stuck at low	SP-COP2 IQ1 (Output) Stuck at high
10	reserved	SP-X[13] output data state	SP-X[13] input data state	SP-X[13] configuration state	don´t care	reserved	SP-X[13] internal module state	don´t care
11	reserved	SP-X[14] output data state	SP-X[14] input data state	SP-X[14] configuration state	don´t care	reserved	SP-X[14] internal module state	don´t care
8 + 4•n	reserved	SP-X[n] collective error fast shut-off	SP-X[n] power supply for Q1 to Q4	SP-X[n] configuration state	don´t care	SP-X[n] external module state	SP-X[n] internal module state	don´t care
9 + 4•n	SP-X[n] output data state	SP-X[n] input data state	reserved	reserved	SP-X[n] I7/I8 two channel	SP-X[n] I5/I6 two channel	SP-X[n] I3/I4 two channel	SP-X[n] I1/I2 two channel
10 + 4•n	SP-X[n] I8 test pulse match	SP-X[n] I7 test pulse match	SP-X[n] I6 test pulse match	SP-X[n] I5 test pulse match	SP-X[n] I4 test pulse match	SP-X[n] I3 test pulse match	SP-X[n] I2 test pulse match	SP-X[n] I1 test pulse match
11 + 4•n	SP-X[n] Q4 Stuck-at low	SP-X[n] Q4 Stuck-at high	SP-X[n] Q3 Stuck-at low	SP-X[n] Q3 Stuck-at high	SP-X[n] Q2 Stuck-at low	SP-X[n] Q2 Stuck-at high	SP-X[n] Q1 Stuck-at low	SP-X[n] Q1 Stuck-at high

8.5 Supported Assembly data

Assemblies are collections of data attributes and are optimized for performance and small telegram overheads. The SP-COP2-ENI supports a set of predefined, static assembly instances for input and output directions. They are accessible via different instances of the CIP Assembly objects. They can be accessed, both with Implicit and Explicit Messaging. The assembly size is variable. In other words, it is possible to request a subset of the complete length of the assemblies. The following table (*Overview of Assembly data bytes in SP-COP2-ENI [ch. 8.5.1, p. 95]*) lists an overview of the supported assembly instances and the meaning of the transported data.

8.5.1 List of Assembly data

Table 58: Overview of Assembly data bytes in SP-COP2-ENI

Instance	Byte	Access	Data type	Meaning	Size	Data range
37	0 to 49	set, get	BYTE[50]	Logic Output Bytes, configured by **Input data block 1 to 5** in samosPLAN 5+ (*more [ch. 8.5.2, p. 97]*)	1 to 50 Byte	0 to 0xff
138	10 to 49	set, get	BYTE[40]	Logic Output Bytes, configured by **Input data block 2 to 5** in samosPLAN 5+ (*more [ch. 8.5.2, p. 97]*)	1 to 40 Byte	0 to 0xff
139	20 to 49	set, get	BYTE[30]	Logic Output Bytes, configured by **Input data block 3 to 5** in samosPLAN 5+ (*more [ch. 8.5.2, p. 97]*)	1 to 30 Byte	0 to 0xff
140	30 to 49	set, get	BYTE[20]	Logic Output Bytes, configured by **Input data block 4 to 5** in samosPLAN 5+ (*more [ch. 8.5.2, p. 97]*)	1 to 20 Byte	0 to 0xff
141	40 to 49	set, get	BYTE[10]	Logic Output Bytes, configured by **Input data block 5** in samosPLAN 5+ (*more [ch. 8.5.2, p. 97]*)	1 to 10 Byte	0 to 0xff
57	0 to 49	get	BYTE[50]	Logic Input Bytes, configured by **Output data set 1** in samosPLAN5+ (*more [ch. 8.5.3, p. 98]*)	1 to 50 Byte	0 to 0xff
	50 to 66	get	BYTE[17]	Reserved	1 to 17 Byte	0 to 0xff
167	0 to 49	get	BYTE[50]	Logic Input Bytes, configured by **Output data set 1** in samosPLAN5+ (*more [ch. 8.5.3, p. 98]*)	1 to 50 Byte	0 to 0xff
	50	get	BYTE	Bit 7: Input Status Bit 6: Output Status Bit 5: Error code ≠ 0	1 Byte	0x00, 0x40, 0x80, 0xc0
	51	get	BYTE	System Mode	1 Byte	1, 2, 3, 4, 5, 7, 21
	52 to 111	get	BYTE[60]	SP-COP Status Bytes (*Instance 3 of Class 120 [ch. 8.4.8.3, p. 92]*), **Output data set 3** (*more [ch. 8.5.3, p. 98]*)	60 Byte	0 to 0xff

The data type of supported assemblies is BYTE which means strings of 8 bits each. The naming in Logix Designer is SINT, which has the same size of 8 bits each.

If PLC requires a configuration assembly, any value or blank can be used for **assembly instance**. **Size** of the configuration assembly shall be zero.

Please use assembly instances from Table *Overview of Assembly data bytes in SP-COP2-ENI [ch. 8.5.1, p. 95]* for **Input** and **Output**. These settings can be used in generic EtherNet module configuration in Logix Designer which can be seen in the illustration below.

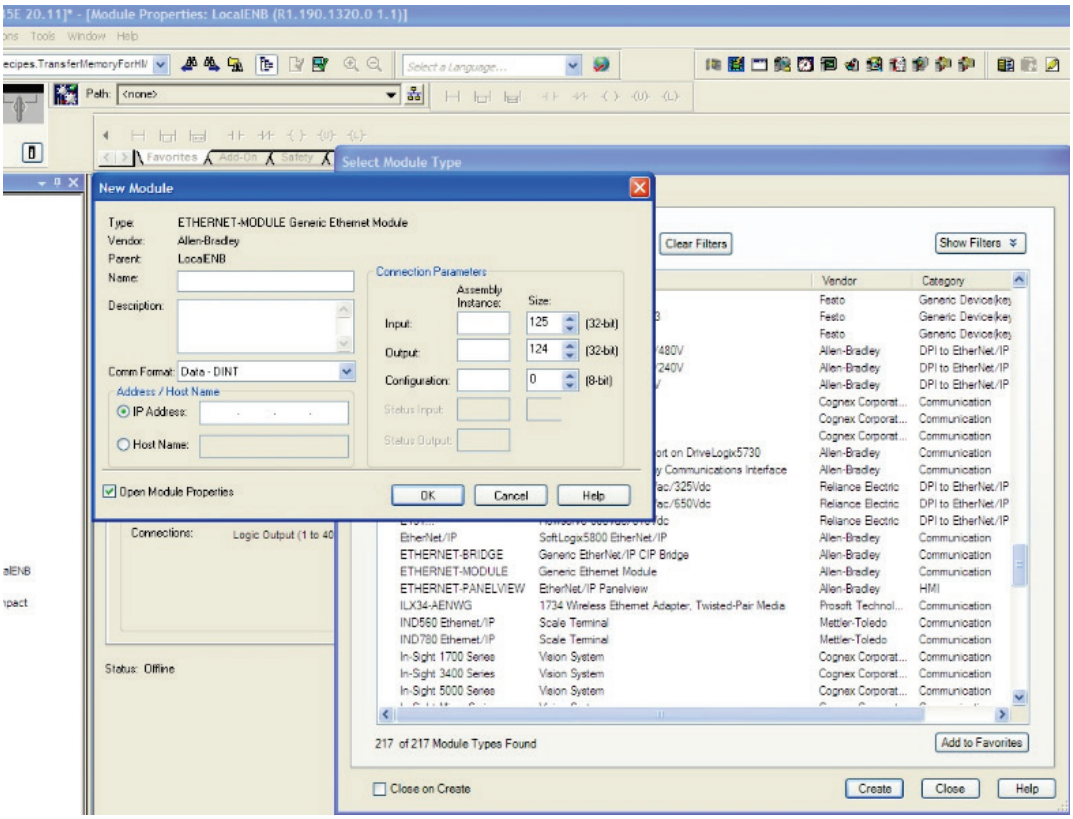


Illustration 29: Generic Ethernet module configuration

8.5.2 Assembly Instances for Logic Output Bytes

8.5.2.1 Assembly Instance 37 = 0x25

The assembly instance 37 belongs to the device profile **General Purpose Discrete I/O Device**. It is a data consumer (O→T) for up to 50 byte.

The first byte of assembly instance 37 is the first byte of the SP-COP2-ENI Logic data, called **Input data block 1 to 5** in samosPLAN5+. The following 49 bytes have linear data mapping in similar manner.

8.5.2.2 Assembly Instances 138 = 0x8a to 141 = 0x8d

The Assembly Instances 138 to 141 are specified to provide more than one output data connection. Output data from the PLC to the SP-COP can be sent in Class 1 connections by Exclusive Owner only. For instance, if one PLC "owns" assembly instance 138, it "owns" output bytes 10 to 49. However, output bytes 0 to 9 are free and can be "owned" by another PLC Exclusive Owner connection (O→T).

Another example is that first PLC "owns" 10 output bytes of assembly instance 37, the second PLC "owns" 10 output bytes of assembly instance 138 and the third PLC "owns" assembly instance 139 with 30 output bytes. Here, three PLCs have Exclusive Owner connections with output data and in general, up to 5 PLCs can share the output data area, 10 bytes each.

Assembly instance 138 is a data consumer for up to 40 bytes, assembly instance 139 is a data consumer for up to 30 bytes, assembly instance 140 is a data consumer for up to 20 bytes and assembly instance 141 is a data consumer for up to 10 bytes.

The first byte of assembly instance 138 is 11th byte of the SP-COP2-ENI Logic data, called **Input data block 2** in samosPLAN5+. The first byte of assembly instance 139 is the 21st byte of the SP-COP2-ENI Logic data, called **Input data block 3** in samosPLAN5+. The first byte of assembly instance 140 is the 31st byte of the SP-COP2-ENI Logic data, called **Input data block 4** in samosPLAN5+. Finally, the first byte of assembly instance 141 is the 41st byte of the SP-COP2-ENI Logic data, called **Input data block 5** in samosPLAN5+.

Write requests are rejected if the assembly is linked to an active I/O connection.

8.5.3 Assembly Instances for Logic Input Bytes

8.5.3.1 Assembly Instance 57 = 0x39

The assembly instance 57 belongs to the device profile **General Purpose Discrete I/O Device**. It is a data producer (T→O) for up to 67 byte.

The first 50 bytes of assembly instance 57 correspond to the first byte of the SP-COP-2-ENI Logic data, called **Output data set 1** in samosPLAN5+. The meaning of bytes 50 to 66 is described here: *Data of Class 1 Connection "Logic Output (1 to 400) and Logic/Physical Input" [ch. 8.3.1, p. 73]*

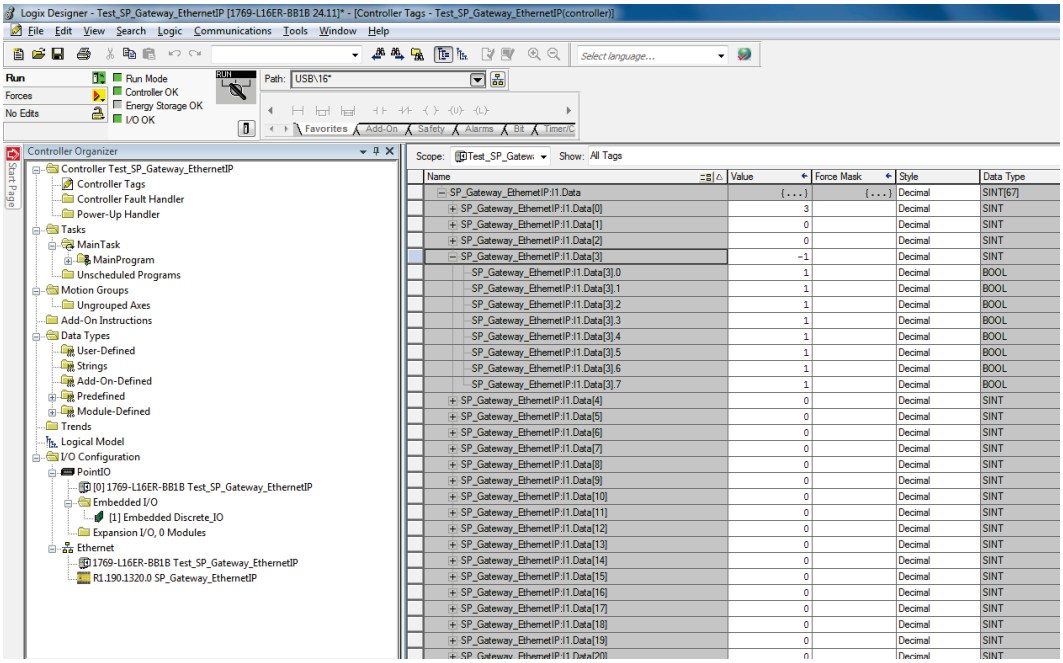


Illustration 30: Example of visualization of assembly instance 57 in Logix Designer

8.5.3.2 Assembly Instances 167 = 0xa7

The Assembly Instance 167 is specified to have an alternative data structure to instance 57. The instance 167 matches the data produced by SP-COP in a more interesting manner.

Assembly instance 167 in SP-COP is a data producer (T→O) for up to 112 byte.

8.5.3.2.1 Byte 0 to 49

The first byte of assembly instance 167 is the first byte of the SP-COP2-ENI Logic data, called **Output data set 1** in samosPLAN5+. The following 49 bytes have linear data mapping in a similar manner.

This is the same mapping of attributes 1 to 50 that the instance 57 has.

8.5.3.2.2 Byte 50

Bit 7 of byte 50 of assembly instance 167 has the same value as the class 29 instance 1 attribute 5, which represents the group status of all physical inputs.

Bit 6 of byte 50 of assembly instance 167 has the same value as the class 30 instance 1 attribute 5, which represents the group status of all physical outputs.

Bit 5 of byte 50 of assembly instance 167, if set, indicates that an error code in class 120 instance 6 attribute 1 is non-zero.

Bits 0 to 4 of byte 50 of assembly instance 167 byte are reserved for future use.

8.5.3.2.3 Byte 51

Byte 51 of assembly instance 167 delivers the system mode of SP-COP. It has the same value as class 120, instance 5 attribute 1.

8.5.3.2.4 Byte 52 to 111

Byte 52 to 111 of assembly instance 167 provides dedicated status bytes of the SP-COP. They have the same values as class 120, instance 3, attributes 1 to 60.

8.6 Accessing to CIP objects

8.6.1 Explicit Messaging

Explicit Messaging uses the TCP/IP protocol and an EtherNet/IP specific encapsulation layer. Explicit Messaging is available as unconnected (UCMM) and connected, e.g. session based. The latter is called **Class 3 Messaging**. Both, UCMM and Class 3 use an EPATH to address the desired data. An EPATH consists of the service, class, instance and attribute ID.

Each attribute of all objects defined in the following tables can be accessed with explicit messaging:

- *Overview of Identity Class (0x01) supported by SP-COP2-ENI [ch. 8.4.1, p. 79]*
- *Overview of Assembly Class (0x04) supported by SP-COP2-ENI [ch. 8.4.2, p. 81]*
- *Overview of Discrete Input Point Object (0x08) supported by SP-COP2-ENI [ch. 8.4.3, p. 82]*
- *Overview of Discrete Output Point Object (0x09) supported by SP-COP2-ENI [ch. 8.4.4, p. 84]*
- *Overview of Discrete Input Group Object (0x1D) supported by SP-COP2-ENI [ch. 8.4.5, p. 85]*
- *Overview of Discrete Output Group Object (0x1E) supported by SP-COP2-ENI [ch. 8.4.6, p. 85]*
- *Overview of Vendor specific Object (Wieland Electric, 0x78) supported by SP-COP2-ENI [ch. 8.4.8.7, p. 93]*

Each request must contain a valid EPATH pointing to the desired object/attribute.

8.6.2 Implicit Messaging

For Implicit Messaging, EtherNet/IP uses the UDP/IP protocol, an EtherNet/IP specific encapsulation layer. Implicit Messaging is also called **Transport Class 1**. The PLC can establish a class 1 connection to the SP-COP2-ENI by sending a **Forward_Open** service request. It configures connection details as the input/output data to exchange, the RPI, unicast or multicast connection, etc. Class 1 connections only support assemblies for input / output data exchange or “place holders” to signal data-less heartbeat connections. Configuration assemblies part of the Forward_Open services are accepted, but not processed by SP-COP2-ENI, except for TCP/IP Object (Class 0xF5).

As the connection configuration details are only sent once in the Forward_Open frame, Implicit Messaging is tuned for performance and has less telegram overhead than Explicit Messaging. Assembly Instances have predefined attributes of predefined order. However, the Originator, e.g. the PLC, sets the data size included in the Forward_Open during Class 1 connection set up. This means, only data bytes from the beginning of the instance to the requested size will be exchanged.

SP-COP2-ENI supports 7 static assembly instances. They are listed in table *Overview of Assembly Class (0x04) supported by SP-COP2-ENI [ch. 8.4.2, p. 81]*. All instance data members are hard-coded. Dynamic assembly instances are currently not supported by the SP-COP2-ENI.

I/O assemblies are either input or output types. The following figure visualizes the data flow with usage of several Assembly Instances. Open assemblies are connected by blue lines and vendor specific assemblies by black lines. The SP-COP is illustrated as a dashed rectangle.

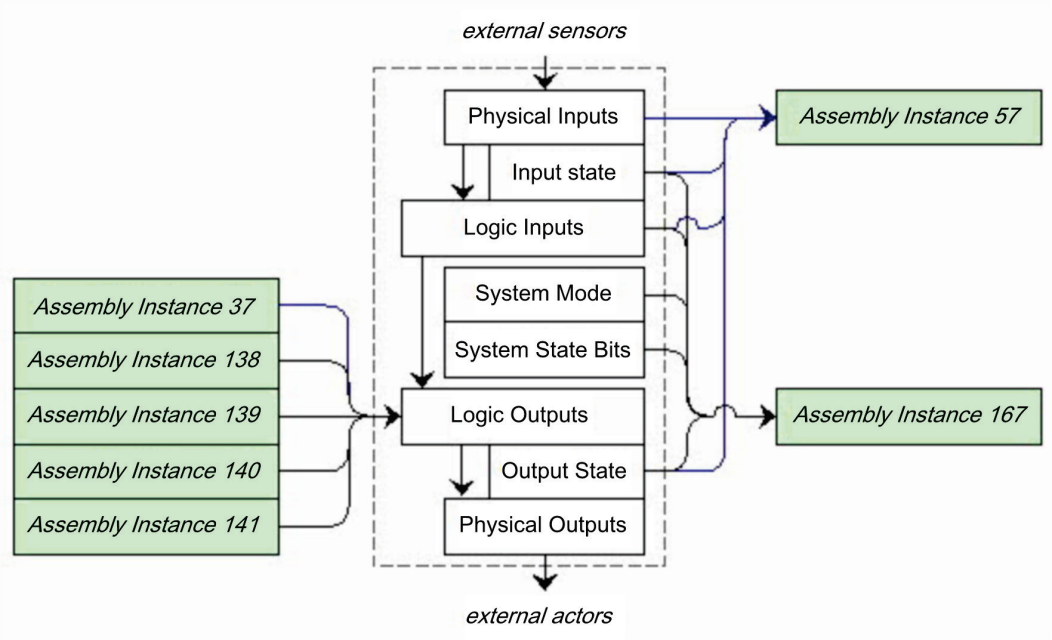


Illustration 31: Data flow with usage of Assembly Instances provides by SP-COP2-ENI

8.6.3 Symbolic Addressing

In addition to addressing assembly instances by choosing connections, symbolic addressing by name can be used.

In samosPLAN5+ the Tag names can be configured in the **Gateway configuration** tab.

Device functionality

Output data (to PLC):

Dataset	Byte	Tag/File-Name	Assembly Instance	Byte
1	0..49	OutDataSet1	57	0..66
2	0..31	OutDataSet2	-	-
3	0..59	OutDataSet3	167	0..111
4	0..59	OutDataSet4	-	-

Input data (from PLC):

Data block	Byte	Tag/File-Name	Assembly Instance	Byte
1	0..9	InDataSet1	37	0..49
2	10..19	InDataSet2	138	10..49
3	20..29	InDataSet3	139	20..49
4	30..39	InDataSet4	140	30..49
5	40..49	InDataSet5	141	40..49

Illustration 32: Configuring symbolic names for assemblies in samosPLAN5+

NOTICE The UCMM (unconnected) Message Client functionality, which can also be configured in samosPLAN5+ is not enabled in module version D-01.01.

8.7 Adjust Performance

Configuring process data byte counts exactly for the application, helps to decrease the amount of data bytes exchanged periodically.

The PLC sets the output byte count in Forward_Open service specific data as **Connection Size** for O→T . The **Fixed/Variable** bit can be set to 1, which mean **Variable**, by PLC.

The PLC sets the input byte count, too. SP-COP will send each RPI, in Forward_Open service specific data as **Connection Size** for T→O. If the **Fixed/Variable** bit is set to 1, which mean **Variable**, by PLC, not all assembly bytes have to be sent.

8.8 Connect with more than one PLC

The EtherNet/IP functionality of the SP-COP2-ENI provides access by more than one PLC. Up to 5 concurrent Encapsulation sessions (input & output) are supported.

If getting process data from SP-COP2-ENI is all that is required, Input only or Listen only connections can be used. Keep in mind, a Listen only connection will be closed automatically by the SP-COP2-ENI if the owner, which has established the Exclusive or Input only connection, closes the connection.

If sending process data to the SP-COP2-ENI from several PLCs, assembly instances 138 to 141 for class 1 connections can be used by other PLCs. Class 3 connections can be established in parallel, if there is no owner conflict. For details see: *List of Assembly data [ch. 8.5.1, p. 95]*

8.9 Diagnostics and troubleshooting

8.9.1 Notifications via network

8.9.1.1 Explicit Message Connection

Device Status is available by reading Class 1, instance 1, attribute 5. Vendor specific interface for alarms and diagnostic for Explicit Message Connections is defined as follows:

Presence of process alarm can be detected by reading class 29, instance 1, attribute 5 and class 30, instance 1, attribute 5. The module mode (**Run** or another state) has to be checked, because the alarm bit is set to 0 = OK every time the module is not in **Run** mode.

The module mode can be detected by reading class 120 Instance 5 Attribute 1.

The presence of module diagnostic events can be detected by reading class 120 Instance 6 Attribute 1.

Detailed reasons for process alarms and system diagnostic events can be found out by reading all 60 attributes of Class 120 Instance 3 which contains the dedicated system status bytes.

8.9.1.2 Implicit Message Connection

If assembly instance 57 is used, Bit 6 and 7 of Byte 66 signal a process alarm when set.

If assembly instance 167 is used, Bit 6 and 7 of Byte 66 signal a process alarm when set. Bit 5 signals diagnostic events or process alarms when set.

Event details can be queried by Explicit Message requests as described here: *Explicit Messaging [ch. 8.6.1, p. 100]*

8.9.2 LED States

8.9.2.1 MS (Module Status)

The SP-COP2-ENI provides one bicolour (Red / Green) LED named **MS**. It is the **Module Status Indicator**.

The Module Status Indicator is *steady off*, if no power is supplied to the device. It *flashes green*, if the device has not been configured. It is *steady Green*, if the device is operating correctly. It *flashes Green / Red*, while the device is performing its power up testing.

The Module Status Indicator *flashes Red*, if EtherNet/IP is activated and the device has detected a Major Recoverable Fault. An incorrect or inconsistent configuration is categorized as Major Recoverable Fault. It is *steady Red* if EtherNet/IP is activated and the device has detected a Major Unrecoverable Fault, which means **Critical Fault**.

Table 59: Selected States of MS LED

Project file	System mode	ext. Error	MS LED state
don't care	Power up	don't care	Green -> Red
deleted	Init	don't care	flashing Green
invalid	Init	don't care	flashing Red
valid	Idle	don't care	flashing Green
valid	Run	No	steady Green
valid	Run	Yes	flashing Green/Red or flashing Red
valid	Critical Fault	don't care	steady Red

8.9.2.2 NET (Network Status)

SP-COP2-ENI provides one bicolour (Red / Green) LED named **NET**. It means **Network Status Indicator**.

The Network Status Indicator is *steady off*, if no power is supplied to the device or is powered on but with no IP address configured (Interface Configuration attribute of the TCP/IP Interface Object). It *flashes Green*, if EtherNet/IP is activated and an IP address is configured, but no CIP connections are established, and an Exclusive Owner connection has not timed out. It is *steady Green*, if an IP address is configured, at least one CIP connection (any transport class) is established, and an Exclusive Owner connection has not timed out. It *flashes Green / Red*, while the device is performing its power up testing.

The Network Status Indicator *flash Red*, if EtherNet/IP is activated and an IP address is configured, and an Exclusive Owner connection for which this device is the target has timed out. The network status indicator is returned to steady green only when all timed out Exclusive Owner connections are re-established. The network status indicator alternates from flashing red to steady green when all connections to the previously timed-out O->T connection points are re-established. Timeout of connections other than Exclusive Owner connections does not cause the indicator to flash red. The Flashing Red state applies to target connections only. Originators and CIP Routers do not enter this state when an originated or routed CIP connection times out.

8.9.2.3 LINK

SP-COP2-ENI provides one Green LED named **LINK**. It is off if no link and is on in case of link.

8.9.2.4 ACT (Activity Status)

SP-COP2-ENI provides one Green LED named **ACT**. It is off if no port activity and is on in case of port activity.

8.9.3 Diagnostics in samosPLAN5+

Additional diagnostics is provided by a log file, named history.csv, on the SD card. Last entries are also accessible via samosPLAN5+ in the tab **Diagnostics**. The timestamp gives the total amount of time the device has been powered on.

When you set the local time via Time **Refresh** button in samosPLAN5+, the **Local time** in the Diagnostics will be set until next power-off.

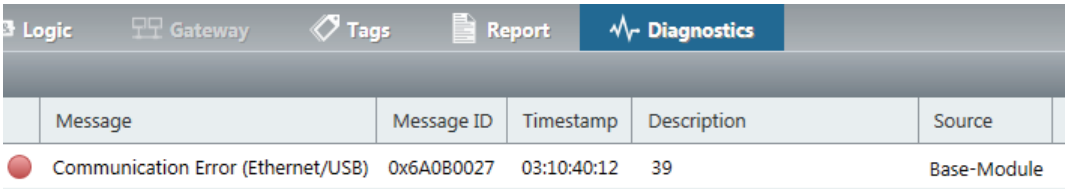


Illustration 33: Diagnostics in samosPLAN5+

Illustration 34: Setting local time via samosPLAN5+

8.10 Abbreviations and Definitions

ACD	Address Collision Detection
ANSI	American National Standards Institute, common used character encoding
AOI	Add On Instruction
AOP	Add On Profile
API	Actual Packet Interval
Attribute	characteristic or feature of an object
Bit	Data unit, having value of 0 or 1
BOOL	Data type specified for CIP devices, means 1 byte value where each of the 8 bit is interpreted independently
Byte, BYTE	Data unit, spanning 8 bit, is unsigned if not other mentioned
CIP	Common Industrial Protocol
Class	Set of objects all of which represent a similar system component. A class is a generalization of the object, a template for defining variables and methods. All objects in a class are identical in form and behavior, but they may contain different attribute values. (Reference: Conformance with CIP (Common Industrial Protocol) Specification, edition 3.18)
CRC	Cyclic redundancy check, a type or its result of hash function for detecting errors in data storage or transmission
Device	Synonym for SP-COP2-ENI
EPATH	Encoded Path, specified for CIP devices
EtherNet/IP	Industrial Ethernet network; combines standard Ethernet technologies with CIP
Gateway	Connection module to industrial networks, like EtherNet/IP, PROFIBUS DB, CANopen, Modbus TCP etc.
ID	unique Identifier
Instance	The actual physical presentation of an object within a class. It identifies one of many objects within the same object class. (Reference: Conformance with CIP (Common Industrial Protocol) Specification, edition 3.18)
IP	Internet Protocol
I/O	Input / Output
LSB	Low Significant Byte
MSB	Most Significant Byte
MPI	Measured Packet Interval, at the moment of measurement this is the API
O→T	Originator to Target
ODVA	Open Device Vendor Association
PC	Personal Computer
PCCC	Programmable Controller Communication Command
PLC	Programmable Logic Controller
RPI	Requested Packet Interval
RX	Receive
SP-COP	samosPRO COMPACT, safety control system consisting of Controller module (SP-COP2-ENI) and optionally pluggable extension gateway and I/O-modules

SP-COP2-ENI	Controller module, contains especially NSC, SC1, SC2 and Safety In- and Outputs
samosPLAN5+	Configuration tool for SP-COP, runs on PC and communicates with NSC
S/N	Serial Number
Stuck-at high	name for unexpected “High“ = “On“ = “1” on a line, unexpected power on line
Stuck-at low	name for unexpected “Low“ = “Off“ = “0” on a line, unexpected non power on line
SHORT_STRING	Data type specified for CIP devices, means character string (1 byte per character, 1 byte length indicator)
T→O	Target to Originator
TCP	Transmission Control Protocol, Internet Standard transport layer protocol defined in RFC 793
TX	Transmit
UCMM	Unconnected Message Manager
UDP	User Datagram Protocol, Internet Standard transport layer protocol defined in RFC 768
UDT	User Defined Tag
UINT	Data type specified for CIP devices, means unsigned 2 byte value
UDINT	Data type specified for CIP devices, means unsigned 4 byte value
USINT	Data type specified for CIP devices, means unsigned 1 byte value
0b	tailing values is in binary format
0x	tailing value is in hexadecimal format
{ }	Array or struct of elements

9 PROFIBUS DP gateway

The following samosPRO COMPACT gateways can be used for PROFIBUS DP:

- SP-PROFIBUS-DP

9.1 Interfaces and operation

Operation and display elements

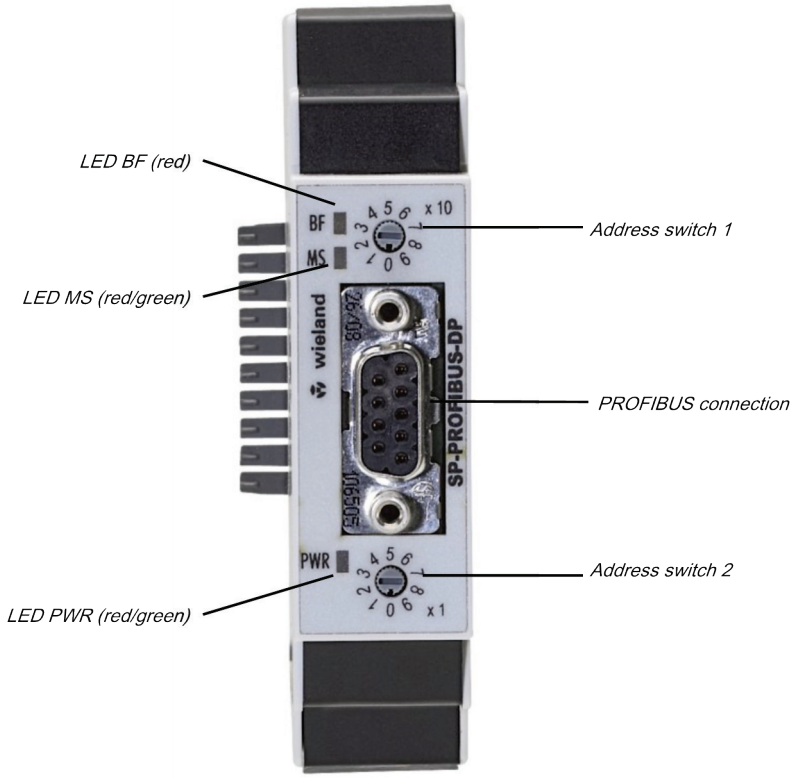


Illustration 35: Operating and display elements of the SP-PROFIBUS-DP

Table 60: Meaning of the state LEDs of the SP-PROFIBUS-DP

LED		Meaning
Key: ○ LED off / ✳ LED flashes / ● LED lights up		
BF	○ Off	Connection to the DP master established
	● Red	No bus connection: Field bus cabling interrupted, address error or the master is no longer transmitting to the bus
MS	○ Off	Power supply switched on, waiting for bus-off
	● Green	Run
	● Green	Stop
	✳ / ✳ Red / green	Run, but the gateway has a fault
	✳ Red	1 Hz: Configuration required or is taking place right now 2 Hz: Critical error on the gateway

LED		Meaning
	Red	Critical error on another module
PWR	Off	No power supply
	Green	Power supply switched on, no error
	Red	Critical errors

Table 61: Address switch of SP-PROFIBUS-DP

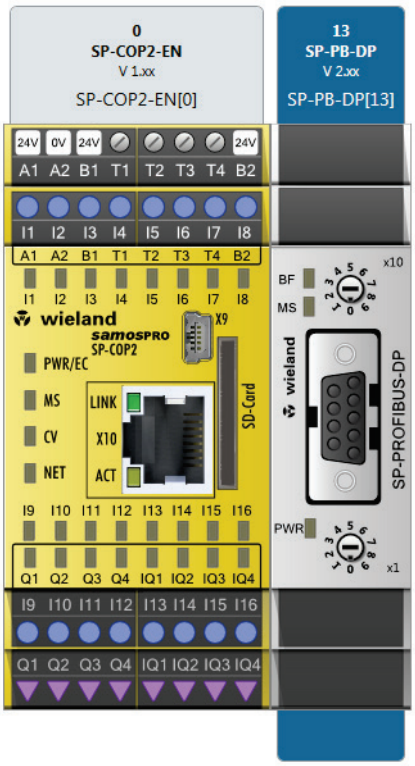
Switches	Function
× 10	Address switch 1 Rotary switch with 10 positions for setting the module address (in tens)
× 1	Address switch 2 Rotary switch with 10 positions for setting the module address (in units)

How to set the PROFIBUS-DP address with the aid of the hardware address switches:

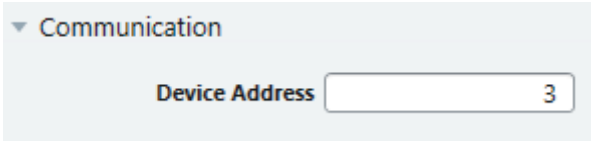
- ➡ Use the hardware address switches at the front of the system to set the PROFIBUS-DP address.
- ➡ Switch the samosPRO COMPACT system off and on again.

How to set the PROFIBUS-DP address with the aid of the software for the samosPLAN5+:

- ➡ Set the two hardware address switches on the front of the device to "00".
- ➡ Start samosPLAN5+.
- ➡ Read in the hardware configuration, including the PROFIBUS-DP gateway. Instructions: "samosPLAN5+ Software" manual, section "Mit dem samosPRO-System verbinden"
- ➡ Change to the **Hardware** view and click on the **SP-PB-DP** module in the work area.



➡ Enter the desired value for the **Control address** parameter under **Communication** in the right side bar.



NOTICE

- You can set an address within the 1 ... 99 range with the aid of the hardware address switches.
- You can set an address within the 3 ... 125 range with the aid of the samosPLAN5+.
- The PROFIBUS master cannot overwrite the address.
- An amended address setting will only become effective once you have switched off the samosPRO COMPACT system and switched it on again.
- In the online mode, you can read out the address set at the PROFIBUS-DP gateway by clicking on the **Read** button above the **PROFIBUS address** field.

Pin assignment

Connection to the PROFIBUS-DP field bus is via a 9-pin D-sub socket.

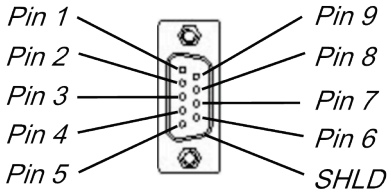


Illustration 36: Pin configuration of D-sub socket and plug for the SP-PROFIBUS-DP

Table 62: Reference for pin configuration

Pin	Description
1	NC
2	NC
3	RxD/TxD-P
4	CNTR-P
5	GND-EXT
6	+5V-EXT
7	NC
8	RxD/TxD-N
9	CNTR-N (GND-EXT)
SHLD	Shielding

Bus cable

The bus topology for PROFIBUS DP is a linear structure consisting of a shielded and twisted 2-lead cable with active bus termination at both ends. The potential bus lengths range from 100 m at 12 Mbit/s to 1200 m at 94 kbit/s.

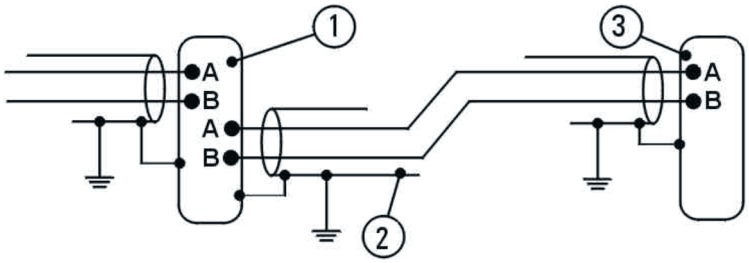


Illustration 37: SP-PROFIBUS-DP bus cable

Table 63: Reference for pin configuration

Position	Description
1	PROFIBUS user gray
2	Shielded bus cable
3	PROFIBUS termination yellow (with integrated terminal resistances)

Line parameters

The bus cable characteristics have been defined in EN 50170 as cable type A.

Table 64: Cable parameters of the SP-PROFIBUS-DP

Characteristic	Value
Wave resistance	135-165 Ω (at a frequency of 3-20 MHz)
Capacity per length unit	< 30 pF/m
Loop resistance	≤ 110 Ω/km
Lead diameter	> 0.64 mm
Wire cross-section	> 0.34 mm^2

These cable parameters provide the following maximum physical dimensions for a bus section:

Table 65: Maximum cable lengths for SP-PROFIBUS-DP

Baud rate (kbit/s)	Maximum cable length (m)
9.6	1200
19.2	1200
93.75	1200
187.5	1000
500	400
1500	200
12000	100

Data transmission rate

The data transmission rate is automatically set. The maximum baud rate is 12 Mbit/s.

9.2 Projecting

GSD file

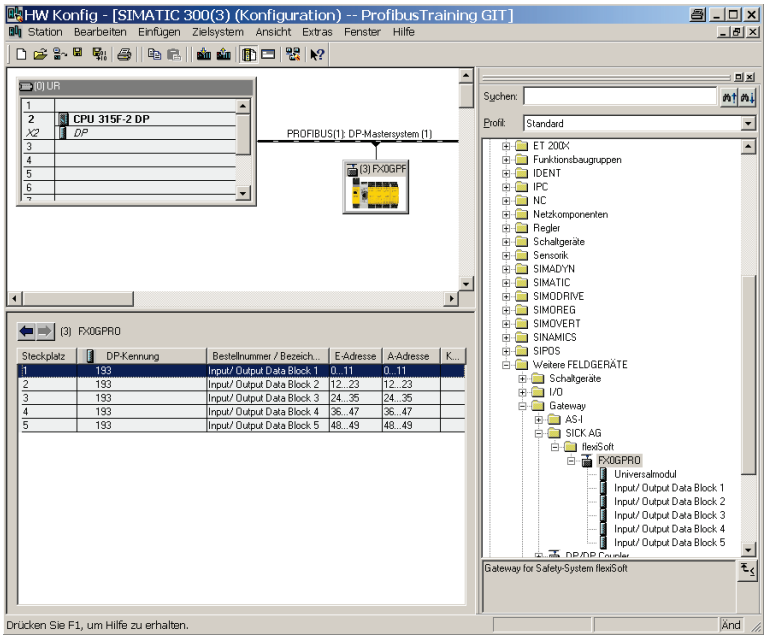
Under normal circumstances, the SP-PROFIBUS-DP is operated on a DP master that reads the device characteristics from the GSD file.

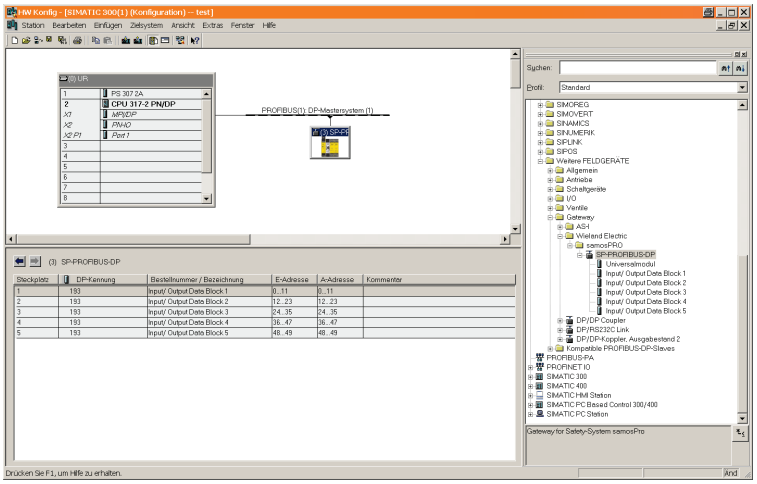
You will find the GSD file and the device symbol for integration into a PLC with PROFIBUS support

- on the Internet on the product side of the SP-PROFIBUS-DP (http://eshop.wieland-electric.com).

Operating data transmitted by the SP-PROFIBUS-DP

The GSD file of the SP-PROFIBUS-DP provides input and output data blocks (virtual I/O device modules), which contain the operating data. These 5 blocks must be projected in a natural sequence (1, 2, 3, 4, 5) in a DP configurator. No other sequence is possible.





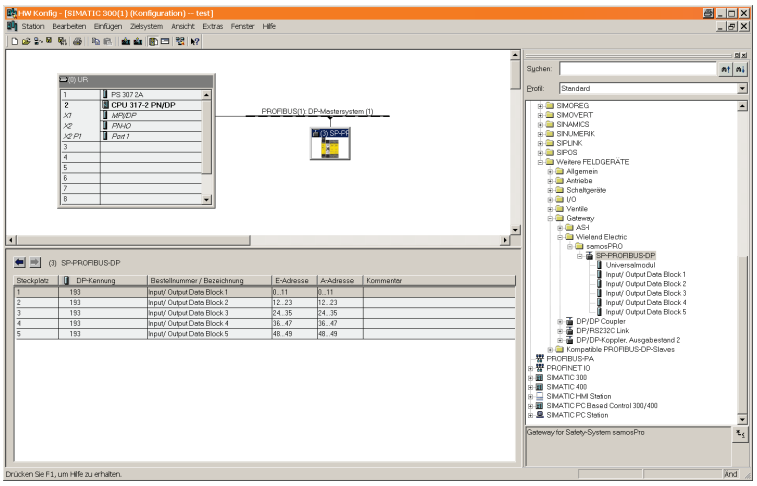


Illustration 38: Example for a PROFIBUS-DP configuration in the Siemens SIMATIC manager

NOTICE

- Depending on the PLC used, further modules may be shown (e.g. "Universal module"). These modules are not required and should be ignored.
- Data blocks 1–4 each contain 12 bytes, while data block 5 contains 2 bytes.

The content of the data blocks can be freely selected, but has been preconfigured as follows for the samosPLAN5+:

Table 66: Predefined content of input data block 1–5 of the SP-PROFIBUS-DP

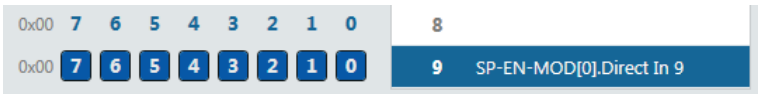
	Data block 1	Data block 2	Data block 3	Data block 4	Data block 5
	Output data block	Output data block	Output data block	Output data block	Output data block
Byte 0	Input values for Module 0 (I1..I8)	Input values for Module 1	Output values for Module 1	Not allocated	Not allocated
Byte 1	Input values for Module 0 (I9..I16)	Input values for Module 2	Output values for Module 2	Not allocated	Not allocated
Byte 2	Input values for Module 0 (IQ1..IQ4)	Input values for Module 3	Output values for Module 3	Not allocated	Not available
Byte 3	Output values for Module 0 (Q1..Q4, IQ1-IQ4)	Input values for Module 4	Output values for Module 4	Not allocated	
Byte 4	Direct data (Off) 0	Input values for Module 5	Output values for Module 5	Not allocated	
Byte 5	Direct data (Off) 1	Input values for Module 6	Output values for Module 6	Not allocated	
Byte 6	Direct data (Off) 2	Input values for Module 7	Output values for Module 7	Not allocated	
Byte 7	Direct data (Off) 3	Input values for Module 8	Output values for Module 8	Not allocated	
Byte 8	Direct data (Off) 4	Input values for Module 9	Output values for Module 9	Not allocated	
Byte 9	Direct data (Off) 5	Input values for Module 10	Output values for Module 10	Not allocated	
Byte 10	Direct data (Off) 6	Input values for Module 11	Output values for Module 11	Not allocated	
Byte 11	Direct data (Off) 7	Input values for Module 12	Output values for Module 12	Not allocated	
Length	12 bytes	12 bytes	12 bytes	12 bytes	2 bytes
Start address	1	13	25	37	49

Detailed information about the content of the process diagram may be found here: *Data transferred to the network (network input data sets [ch. 3.2, p. 14]*).

Delete any bytes not required

You can delete bytes pre-allocated by samosPLAN5+ that you do not require by clicking on them with the mouse.

- ➡ Switch to the **Gateway** view.
- ➡ Click on the byte you do not need and wish to delete.



- ➡ Click on the **Delete** icon in the command bar.

You will find further information about how to configure the process diagram here:

- *Configuration of gateways with samosPLAN5+ [ch. 5, p. 31]*
- "samosPLAN5+ Software" manual (BA000968)

Allocating bytes to other addresses

samosPLAN5+ allocates the addresses by default. You can manually change this address allocation by moving bytes.

In our example, we have shifted **byte 1** to **byte 23** in **tab 1**.

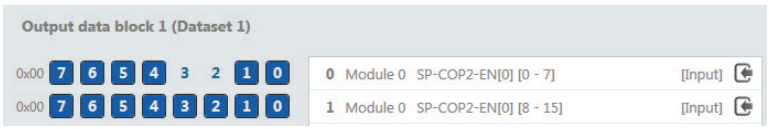


Illustration 39: Initial situation

- ➡ Ensure that the desired address (**byte 23** in our example) has not been allocated.
  In our example, **byte 23** has already been allocated to **module 12**. We therefore first have to delete **byte 23**.

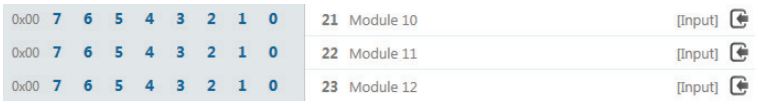


Illustration 40: Byte 23

- ➡ Delete the byte you wish to reallocated (**byte 1** in our example).
  To do this, click on the byte in the work area and click on the **Delete** symbol in the command bar.

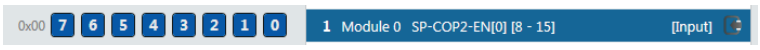


Illustration 41: Delete the byte at its place of origin.

- ➡ Choose the desired module under **Left side bar | Library**.

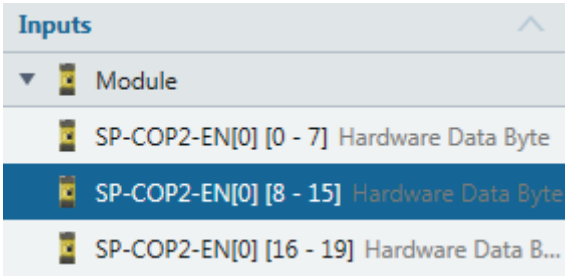


Illustration 42: Module in the library

- ➡ Use the mouse button to drag the module into the work area on **byte 23**.

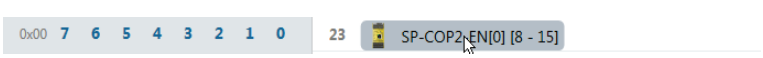


Illustration 43: Module on new byte

9.3 PROFIBUS configuration of the gateway - how the data are transferred

The following steps are required to configure the communication between the PLC and the gateway.

NOTICE

This documentation does not address the installation of the PROFIBUS-DP network or the other components of the automation system project in the network configuration tool. It is assumed that the PROFIBUS project in the configuration program, e.g. the SIEMENS SIMATIC Manager, has already been set up. The examples shown refer to configurations set up with the aid of the SIEMENS SIMATIC manager.

Step 1: Install the device master file (GSD)

Before the SP-PROFIBUS-DP can be used for the first time as part of the network configuration tool, e.g. the SIEMENS SIMATIC Manager, the device master file (GSD) of the gateway must first be installed in the hardware catalog of the tool.

- ➡ Download the GSD file and the equipment symbol from the product site for the SP-PROFIBUS-DP: (http://eshop.wieland-electric.com).
- ➡ Follow the instructions for the installation of GSD files in the online help section or in the user manual for the PROFINET network configuration tool.

If you are using SIEMENS SIMATIC Manager – HW Config, the gateway will subsequently appear in the hardware catalog under >>**PROFIBUS DP** > **Other field equipment** > **Gateway** > **Wieland** > samosPRO COMPACT.

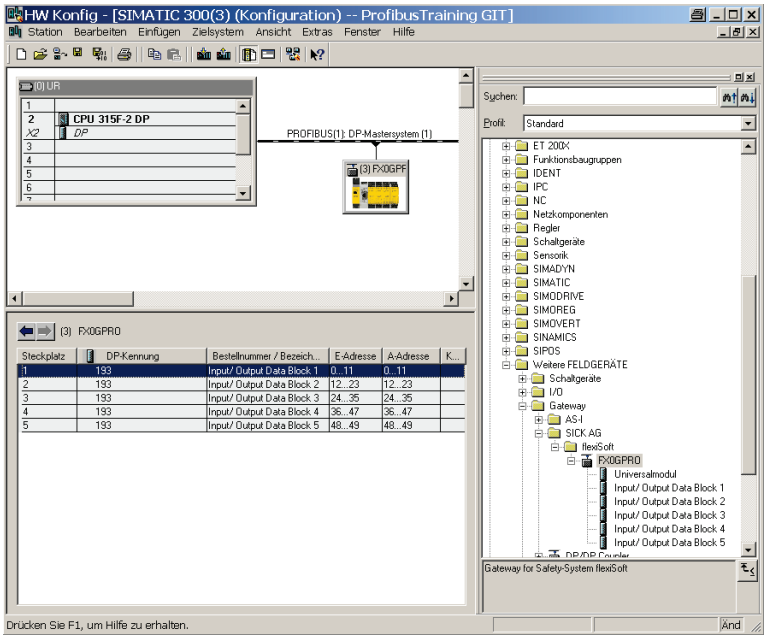
Step 2: Add the gateway to the project

To make the system data for the samosPRO COMPACT system available in the process diagram of the PLC, the gateway must first be added to the hardware configuration. The procedure to be used depends on the hardware configuration software of the PLC used. Please also read the documentation for the corresponding software in this regard.

The example below shows how the gateway is added to a SIEMENS SIMATIC manager project.

In the SIEMENS SIMATIC hardware manager, you will find the gateway in the hardware catalog under >>**PROFIBUS DP** > **Other field equipment** > **Gateway** > **Wieland** > samosPRO COMPACT.

- ➡ Use the drag&drop function to pull the equipment into the PROFIBUS network. Example:



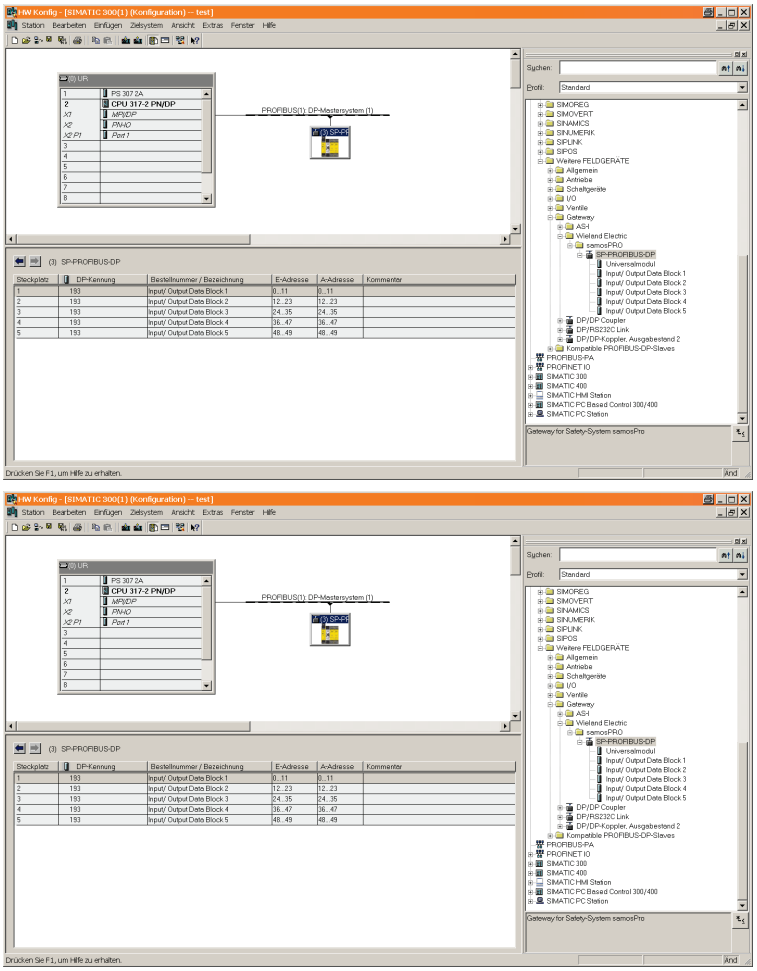


Illustration 44: PROFIBUS-DP gateway in the PROFIBUS HW Config

Diagnostic data for SP-PROFIBUS-DP

The SP-PROFIBUS-DP makes diagnostic data available via PROFIBUS-Standard-DP-V0 diagnosis:

- Standard diagnosis (6 bytes)

9.4 Diagnosis and troubleshooting

Information about the diagnosis of the samosPRO COMPACT system may be found in the "samosPLAN5+ Software" manual (BA000968).

Table 67: Troubleshooting for the SP-PROFIBUS-DP

Error		Possible cause	Possible remedy
Key: ○ LED off / ☀ LED flashes / ● LED lights up			
The samosPLAN5+ cannot establish a link to the samosPRO COMPACT gateway.		The SP-PROFIBUS-DP has no power supply.	• Switch on the power supply. • Check the communication settings in samosPLAN5+.
The SP-PROFIBUS-DP is not providing any data.		• Configuration required. • The configuration has not yet been fully transmitted.	• Configure the SP-PROFIBUS-DP and transfer the configuration to the system. • Wait until the configuration has been fully transferred.
LED PWR	● Green		
LED BF	○ Off		
MS LED	☀ Red (1 Hz)		
The SP-PROFIBUS-DP is not providing any data.		No data set was activated.	Activate at least one data set.
LED PWR	● Green		
LED BF	○ Off		
MS LED	● Green		
The SP-PROFIBUS-DP is not providing any data.		SP-PROFIBUS-DP is in the stop state	• The CPU/application has stopped. • Start the CPU (change to Run mode)
LED PWR	● Green		
LED BF	○ / ● Off / red		
MS LED	☀ Green (1 Hz)		
The SP-PROFIBUS-DP is not providing any data.		PROFIBUS-Master is in stop mode	Set the PROFIBUS-Master to Run mode
LED PWR	● Green		
LED BF	○ Off		
MS LED	● Green		
The SP-PROFIBUS-DP functioned correctly after configuration, but suddenly provides no more data.		• The PROFIBUS hardware address of the SP-PROFIBUS-DP was changed. • The PROFIBUS line has been interrupted.	• Check the PROFIBUS address settings on the hardware. • Check the PROFIBUS cabling. • Check the PROFIBUS master.
LED PWR	● Green		
LED BF	● Red		
MS LED	☀ / ☀ Red / green		

Error		Possible cause	Possible remedy
SP-PROFIBUS-DP is in critical fault.		• Internal equipment error on the SP-PROFIBUS-DP. • The module version of the CPU does not support any samosPRO COMPACT gateways.	• Switch the samosPRO COMPACT system power supply off and on again. • Check the diagnostic messages with the aid of the samosPLAN5+. • Use a CPU with the required module version (see Proper Use). • If the error persists, replace the gateway.
LED PWR	Green		
LED BF	Red		
MS LED	Red (2 Hz)		
The SP-PROFIBUS-DP / samosPRO COMPACT system is in the critical error state.		• The SP-PROFIBUS-DP is not properly connected to the other samosPRO COMPACT modules. • The module connection plug is dirty or damaged. • Another samosPRO COMPACT module has an internal critical error.	• Correctly plug in the SP-PROFIBUS-DP. • Clean the connection plug and socket. • Switch on the power supply again. • Check the other samosPRO COMPACT modules.
LED PWR			
LED BF	Off		
MS LED			

10 CANopen gateway

The following samosPRO COMPACT gateways can be used for CANopen:

- SP-CANopen

10.1 Interfaces and operation

Operation and display elements

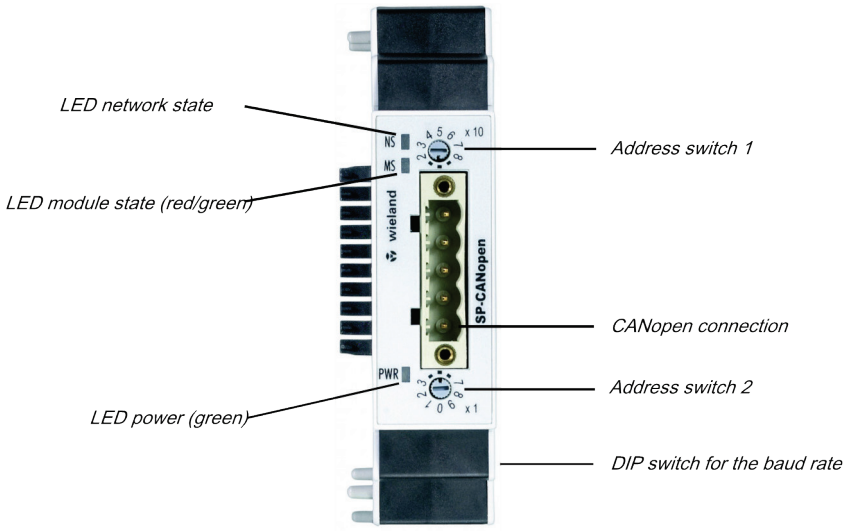


Illustration 45: Operating and display elements of the SP-CANopen

Table 68: Reference: State LEDs of the SP-CANopen

LED		Meaning
Key: ○ LED off / ✳ LED flashes / ● LED lights up		
PWR Power	○ Off	No power supply
	● Green	Ready for operation, power supply switched on
	● Red	System error
NS (Network state)	○ Off	CANopen state: stopped (except for node guarding and heartbeat, when activated)
	● Green	CANopen state: Ready for operation (PDO and SDO data exchange)
	✳ Green	CANopen state: Pre-operational (only SDO data exchange)
	● Red	CAN-Bus off (hardware problem on CAN - physical layer) or error passive
	✳ Red (1 Hz)	Node guarding failed (NMT master no longer monitors the slave) or heartbeat consumer failure
MS (module state)	○ Off	Switch on
	● Green	Executing, SBUS+ and PDO state: all "Good"
	✳ Green	Idle (cable not connected or node guarding failed)

LED		Meaning
	Red / green	Executing, SBUS+ and PDO state: at least one "Bad"
	Red	Critical error, caused by emergency bit
	Red (1 Hz)	Configuration required or is taking place right now
	Red (2 Hz)	Critical error, caused by gateway itself

Further information: *Diagnosis and troubleshooting [ch. 10.12, p. 144]*

How to set the CANopen address with the aid of the hardware address switches

- ➡ Set the CANopen address switches using the hardware address switches at the front of the system.
- ➡ Switch the samosPRO COMPACT system off and on again.

Table 69: Address switch on SP-CANopen

Switches	Function
× 10	Address switch 1 Rotary switch with 10 positions for setting the module address (in tens)
× 1	Address switch 2 Rotary switch with 10 positions for setting the module address (in units)

How to set the baud rate with the aid of the hardware DIP switches:

- ➡ Set the baud rate using the DIP switches on the equipment.
- ➡ Switch the samosPRO COMPACT system off and on again.

Illustration 46: Setting the DIP switches on the SP-CANopen

Table 70: Setting the DIP switches on the SP-CANopen

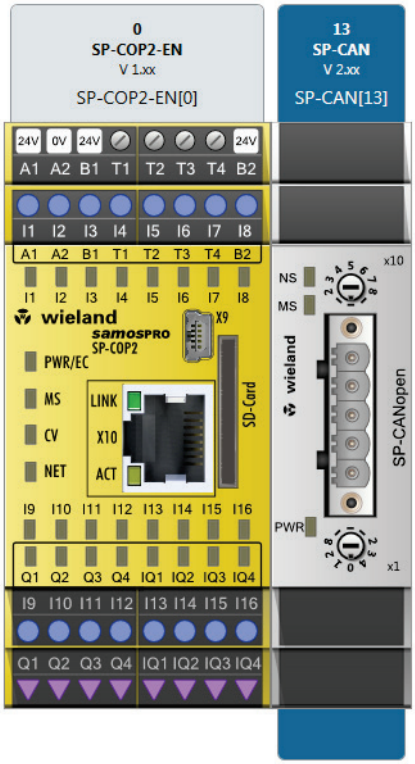
Baud rate (kbit/s)	DIP 1	DIP 2	DIP 3	DIP 4
125	On	On	On	Off
250	Off	On	On	Off
500	On	Off	On	Off
800	Off	Off	On	Off
1000	On	On	Off	Off

NOTICE

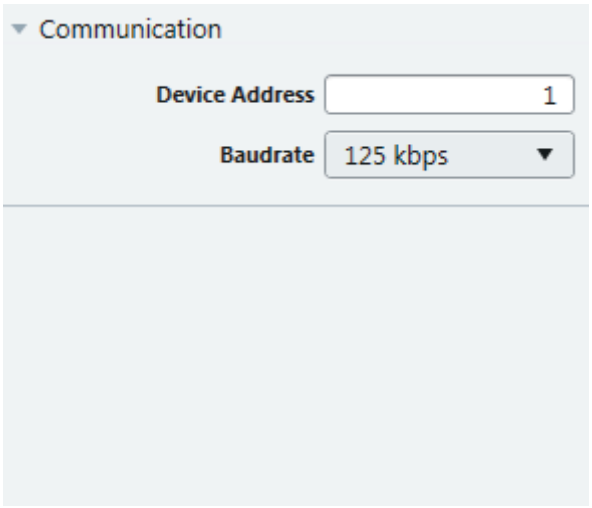
- All other DIP switch settings will set the baud rate to 125 kbit/s.
- When the address switches on the equipment are set to "00", the DIP switch settings are ignored and the baud rate setting in the samosPLAN5+ is used.

How to set the CANopen address and the baud rate with the aid of the software for the samosPLAN5+:

- ➡ Set the two hardware address switches on the front of the device to "00".
- ➡ Start samosPLAN5+.
- ➡ Read the hardware configuration, including the CANopen gateway.
  Instructions: "samosPLAN5+ Software" manual, Section "Connecting to the samosPRO system"
- ➡ Change to the **Hardware** view and click on the **SP-CAN** module in the work area.



- ➡ Enter the desired value for the **Control address and baud rate** parameters under **Communication** in the right side bar.



- ➡ Combine samosPLAN5+ with the samosPRO system and transmit the configuration.
  More detailed information on the link to the controls: "samosPLAN5+ Software" manual (BA000968), section "Combine with the samosPRO system"

NOTICE

- You can set an address within the 1 ... 99 range with the aid of the hardware address switches.
- You can set an address within the 1 ... 127 range with the aid of the samosPLAN5+.
- The CAN-open master cannot overwrite the address.
- When the CANopen address and the baud rate are set with the aid of the samosPLAN5+, the settings become valid immediately after transferring the configuration (i.e. without first switching the samosPRO COMPACT system off and on again). Exception: When the system is in the Bus-Off state, a power cycle is required.

Pin assignment

The connection to the CANopen field bus takes place with the aid of a 5-pin open-style plug.

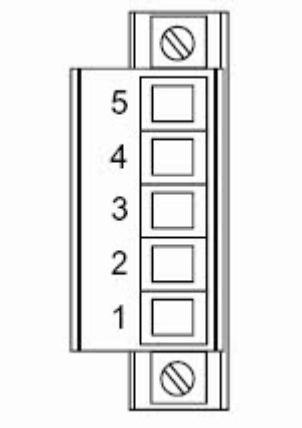


Illustration 47: Open-style plug on SP-CANopen

Table 71: Reference: Allocation of open-style plug on SP-CANopen

Pin	Description	
5	–	–
4	H CAN_H	CAN High
3	DR (CAN_SHLD)	Shielding connection (optional)
2	L CAN_L	CAN Low
1	–	–

Bus cable

CANopen is based on a linear topology with shielded, two-lead twisted-pair cables and terminal resistances at both bus ends. The shielding is connected to ground at both ends. The transmission rate depends on the network length and ranges from 125 kbit/s to 1000 kbit/s. The potential network lengths range from 20 m at 1000 kbit/s to 500 m at 125 kbit/s.

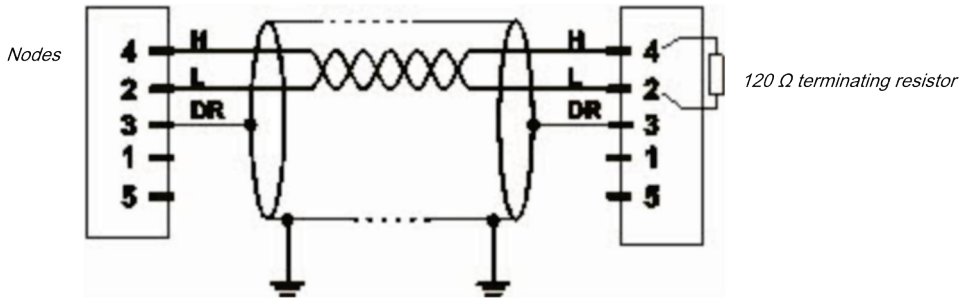


Illustration 48: CANopen bus cable

NOTICE

It is not necessary to connect a power supply (Pin 1/5) to the SP-CANopen.

The following maximum physical values are possible:

Table 72: Maximum cable lengths for SP-CANopen

Bit rate (kbit/s)	Maximum cable length (m)
125	500
250	250
500	100
800	40
1000	20

EDS file

The equipment characteristics are described with the aid of the electronic data sheet (EDS) file), that makes use of any standard bus configuration tool.

You will fin the EDS file and the equipment symbol for integration into a PLC of the product website of the SP-CANopen on the Internet (http://eshop.wieland-electric.com).

10.2 CANopen configuration of the gateway - how the data are transferred

NOTICE

This documentation does not address the installation of the CANopen network or the other components of the automation system project in the network configuration tool. It is assumed that the CANopen project in the configuration program, e.g. 3S Software CoDeSys 2.x, has already been set up. The examples shown refer to configurations set up with the aid of CoDeSys 2.3.

The following steps are required to configure the communication between the PLC and the gateway.

Step 1: Install the electronic data sheet (EDS file)

Before the SP-CANopen can be used for the first time as part of the network configuration tool, e.g. CoDeSys 2.3, the electronic data sheet (EDS file) of the gateway must first be installed in the hardware catalog of the tool.

- ➡ Download the EDS file and the equipment symbol from the product site for the SP-CANopen: (http://eshop.wieland-electric.com).
- ➡ Follow the instructions for the installation of EDS files in the online help section or in the user manual for the CANopen network configuration tool.

Example – How to install the EDS file with CoDeSys 2.3:

- ➡ Open the window for editing the **control configuration**.

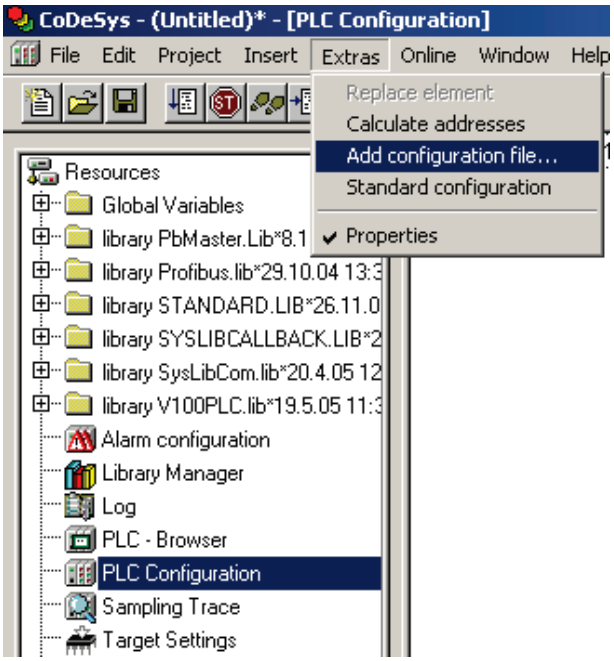


Illustration 49: CoDeSys editing window for control configuration

- ➡ Choose the command **Add configuration file...** from the Extras menu. A file selection window is opened.
- ➡ Select the EDS file for the SP-CANopen and click on the **Open** button.

Step 2: Add the gateway to the controls

To make the system data for the samosPRO COMPACT system available in the process diagram of the PLC, the gateway must first be added to the hardware configuration. The procedure to be used depends on the hardware configuration software of the PLC used. Please also read the documentation for the corresponding software in this regard.

- ➡ Open the window for editing the **control configuration** and select the controls.

➡ Click on the controls using the right mouse button and open the **Insert** menu.

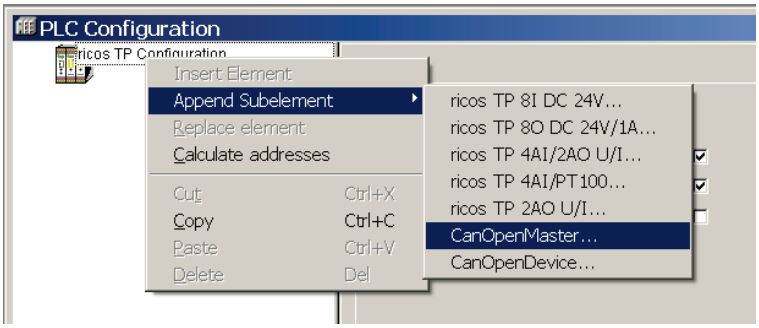


Illustration 50: Attaching a CanMaster with CoDeSys 2.3

➡ Select the command **CanMaster ....** from one of the two menus under **Attach sub-element.** A CanMaster will be attached to the controls.
➡ Now select the CanMaster.
➡ Click on the CanMaster using the right mouse button or open the **Insert** menu.

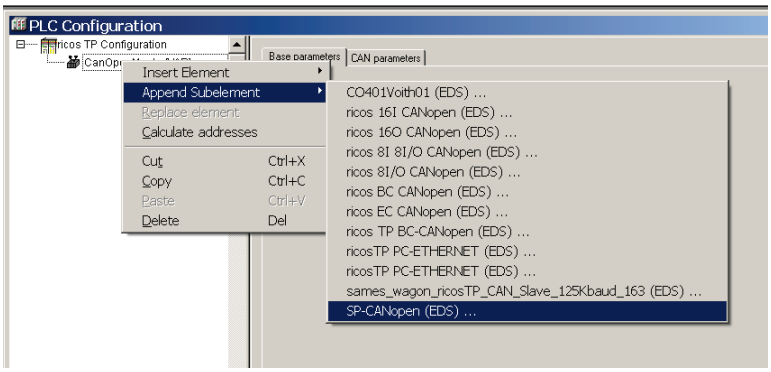


Illustration 51: Attaching the SP-CANopen with CoDeSys 2.3

➡ Select the command **SP-CANopen00000 (EDS)** from one of the two menus under **Attach sub-element** to attach the SP-CANopen to the CanMaster.

Step 3: Select and configure the process data objects (PDOs)

Once you have added the device to the automation network, you must configure the process data objects to be used and how to transfer them.

Example – How to install the PDO transmission type with CoDeSys 2.3:

➡ Select SP-CANopen from the control configuration edit window. Then click on the Send PDO mapping index card on the right.

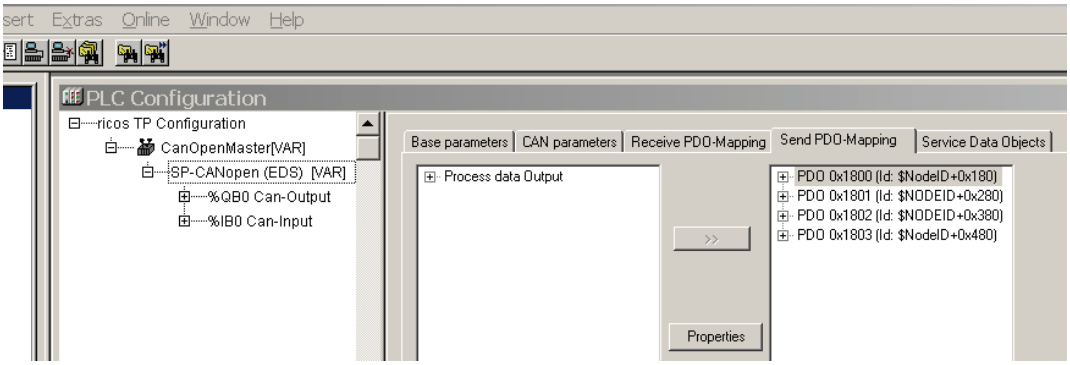


Illustration 52: PDO configuration with CoDeSys 2.3

➡ Select one of the PDOs shown (e.g. PDO 1) and click on the Properties button. The PDO Properties dialog window will open.

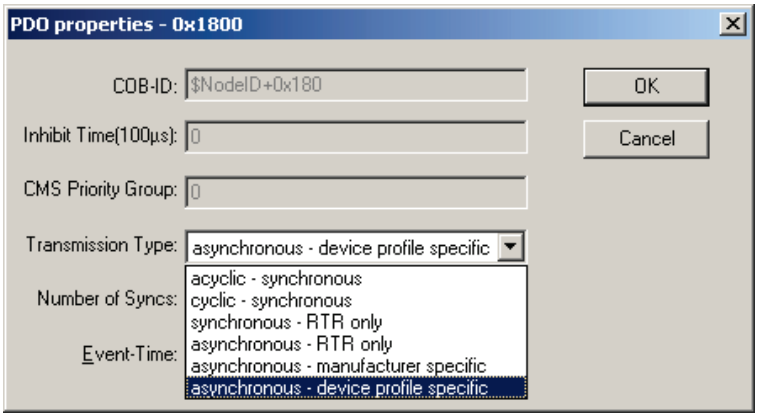


Illustration 53: PDO Properties dialog window in CoDeSys 2.3

➡ From the selection, choose the desired transmission type for the PDO, enter the event time in ms and click on OK. More detailed information in this regard may be found in the section "Transmission types for the TxPDOs" on page 107 and in the manual for your CanOpen configuration software.

➡ Repeat these steps for the other transmission and receiving PDOs.

10.3 CANopen configuration of the gateway - which data are transferred

Each CANopen device stores its data in objects listed in the object directory. The service data objects (SDOs) mainly contain the CANopen configuration data, while the process data are stored in process data objects (PDOs). Communication objects are used to read and write these SDOs and PDOs and to control the devices. The following sections contain more detailed descriptions of the various objects.

Predefined Connection Set (PCS)

The Predefined connection set provides a simple CAN identifier structure. The SP-CANopen gateway provides communication objects that can be addressed or transmitted with the aid of this CAN identifier. The PCS consists of 2 broadcast objects (NMT and SYNC) and a total of 12 peer-to-peer objects. Each of these objects has a clear 11-bit CAN identifier, which consists of a function code and a device address. The device address for the broadcast objects is 0, while that for the other objects is within the range of 1 ... 127.

Table 73: Structure of the CAN identifiers

Bit number										
10	9	8	7	6	5	4	3	2	1	0
Function code				Device address						

Table 74: PCS communication objects

Object	CAN identifier	Meaning
Broadcast objects		
Peer-to-peer objects		
NMT	00h	Network management
SYNC	80h	Sync message
EMERGENCY	081h…0FFh	State message
TxPDO1	181h…1FFh	Send process data object 1
RxPDO1	201h…27Fh	Receive process data object 1
TxPDO2	281h…2FFh	Send process data object 2
RxPDO2	301h…37Fh	Receive process data object 2
TxPDO3	381h…3FFh	Send process data object 3
RxPDO3	401h…47Fh	Receive process data object 3
TxPDO4	481h…4FFh	Send process data object 4
RxPDO4	501h…57Fh	Receive process data object 4
TxSDO	581h…5FFh	Send service data project
RxSDO	601h…67Fh	Receive service data object
NMT-ErrorControl	701h…77Fh	Node guarding

Each object starts with a CAN identifier, followed by a RTR bit (remote transmission request), followed by a data length code (DLC), followed by 0 to 8 data bytes. The DLC (4 bits) provides the number of data bytes.

10.4 NMT – network management

The broadcast object NMT is used to start, stop or initialize CANopen devices. A device in the CANopen network must take on the role of the NMT master for this purpose. This is usually the PLC. All other devices are regarded as NMT slaves. NMT services are broadcast services to which the slaves do not generate responses.

All NMT objects start with the CAN-ID 00h.

Broadcast service for an NMT slave with the address N:

Table 75: Network management for an NMT slave with the address N

CAN-ID	DLC	DATA							
00h	2	OP	N						

Broadcast service for all NMT slaves:

Table 76: Network management for all NMT slaves:

CAN-ID	DLC	DATA							
OP	NMT command			Explanation					
00h	2	OP	0						
80h	Go to "Pre-Operational"			After booting, an NMT slave will automatically go into the pre-operational state. In this state, communication via SDOs is permitted, but not via PDOs. The NMT slave can be set to this state from another state.					
01h	Go to "Operational"			The operational state is reached from the "pre-operational" state. Communication via PDOs is possible in this state and the CANopen slave responds to sync commands. **Note:** During the transition to the NMT operational state, each slave sends a TxPDO with the transmission type = 255, so that the NMT master is informed about the current input configuration.					
02h	Go to "Prepared/Stopped"			Communication via SDO or PDO is not possible in this state and the device also does not respond to sync commands.					
81h	Go to "Reset node"			This will trigger a re-initialization of the CANopen function in the NMT slave.					
82h	Go to "Reset communication"			This will trigger a re-initialization of the CANopen function in the NMT slave; the toggle bit for node guarding is set to 0.					

Example for resetting all communication:

The following NMT object (CAN-ID = 00h) contains 2 data bytes (DLC = 2). Data byte 1 contains the command "Reset communication" (82h), data byte 2 addresses this command to all devices in the CANopen network (address = 0):

Table 77: Example of an NMT object for resetting all communication

CAN-ID	DLC	DATA							
00h	2	82h	0						

10.5 SYNC

The SYNC command results in all TxPDOs of a CANopen slave being sent. It is thus possible to prompt the slave with the aid of SYNC.

Table 78: Prompting of inputs with the aid of SYNC

CAN-ID	DLC	DATA							
80h	0								

The slave sends all input values when he receives this command. All TxPDOs are sent.

To ensure that the slave automatically sends the current input values when receiving a SYNC command, the transmission type for the relevant PDOs must be set to 1 (cyclic, synchronous). In addition, the device must be in the "operational" state.

It is possible to amend the transmission type for the TxPDOs with the aid of the SDOs 1800 … 1803 (PDO communication parameter) and to amend Sub-Object 2. The following types are permitted:

- Acyclic/synchronous = 0
- Cyclic/synchronous = 1 = 1 ... 240
- Acyclic once device profile = 255 (only for TxPDO 1 … 4, digital inputs)

10.6 Node guarding

An NMT master (e.g. a PLC with integrated CANopen master) only makes use of the NMT error control object to detect the failure of an NMT slave with the

address N. The NMT slave must respond to the query of the NMT master within the node guarding time. The node guarding time must be monitored by the NMT master.

The NMT master sends a CAN message with the identifier <700h + node ID> and RTRBit (remote transmission request).

Query of NMT master:

Table 79: Query of NMT master

CAN-ID	RTR	DLC	DATA							
700h + N	1	0								

The slave (e.g. the SP-CANopen) then sends a status byte 1 with the following content:

Response of the slave:

Table 80: Response of the slave

CAN-ID	DLC	DATA							
700h + N	1	Byte1							

Table 81: Remote transmission request

Bit	Meaning	
7	Toggle bit changes its value between two consecutive queries	
6…0	NMT status	4 = Stopped 5 = Operational 127 = Pre-operational

Bootup

During the bootup, the gateway sends a bootup message with the CAN-ID 700h+N, DLC = 1 and Byte 1 = 0.

Heartbeat producer

When the gateway has been configured as a heartbeat producer (i.e. when SDO 1017 contains a value for the producer heartbeat time, see table *"Supported SDOs" [ch. 10.9, p. 134]*), then sends a cyclical message with the CAN-ID 700h+N, DLC = 1 and Byte 1 = 05h. The toggle bit (bit 7) is always 0.

Heartbeat consumer

When the gateway has been configured as a heartbeat consumer (i.e. when SDO 1016.1 contains a value for the consumer heartbeat time, see table *"Supported SDOs" [ch. 10.9, p. 134]*), then at least one node guarding message must be received within the configured consumer heartbeat time (typically from a NMT master).

10.7 PDO communication

Process data objects (PDOs) are the real-time objects of the CANopen field bus. They are sent without a protocol overhead, i.e. the receiver sends no confirmation.

The SP-CANopen provides four transmit process data objects (TxPDOs), which contain the operating

data to be sent to the network and four receive process

data objects (RxPDOs) for the operating data received from the network.

CANopen objects are addressed with the aid of 11-bit CAN identifiers. As a pre-set, the CAN identifier derives each object from the object type and the configured CANopen device address. The CAN identifier of the PDOs can be changed by using SDOs 1400 to 1403 for the RxPDOs and SDOs 1800 to 1803 for the TxPDOs ("PDO linking").

NOTICE Each process data object contains 8 bytes.

The content of the process data objects can be freely selected, but has been preconfigured as follows for the samosPLAN5+:

Table 82: Pre-sets for the content of the transmit process data objects (TxPDOs) of the SP-CANopen

	PDO#1	PDO#2	PDO#3	PDO#4
	Output data - Block 1	**Output data - Block 2**	**Output data - Block 3**	**Output data - Block 4**
Byte 0	Input values for Module 0 (I1..I8)	Input values for Module 1	Input values for Module 9	Output values for Module 5
Byte 1	Input values for Module 0 (I9..I16)	Input values for Module 2	Input values for Module 10	Output values for Module 6
Byte 2	Input values for Module 0 (IQ1..IQ4)	Input values for Module 3	Input values for Module 11	Output values for Module 7
Byte 3	Output values for Module 0 (Q1..Q4, IQ1-IQ4)	Input values for Module 4	Input values for Module 12	Output values for Module 8
Byte 4	Direct data (Off) 1	Input values for Module 5	Output values for Module 1	Output values for Module 9
Byte 5	Direct data (Off) 2	Input values for Module 6	Output values for Module 2	Output values for Module 10
Byte 6	Direct data (Off) 3	Input values for Module 7	Output values for Module 3	Output values for Module 11
Byte 7	Direct data (Off) 4	Input values for Module 8	Output values for Module 4	Output values for Module 12

Detailed information about the content of the process diagram may be found here:
Configuring the gateway output values (tab 1) [ch. 5.3, p. 40]

You will find further information about how to configure the process diagram here:

- *Configuration of gateways with samosPLAN5+ [ch. 5, p. 31]*
- "samosPLAN5+ Software" manual (BA000968)

NOTICE

- The process data can also be written and read with the aid of service data objects SDO 6000 and SDO 6200 (see *SDO communication [ch. 10.8, p. 132]*). Easy access via SDO is recommended for diagnostic purposes. More rapid PDO communication is to be used for normal operation.
- After starting up or changing the configuration (either with the aid of the CANopen master or with samosPLAN5+), the LED MS of the CANopen gateway flashes red/green until an initial transmit/receive data exchange has taken place via PDO or SDO 6000/SDO 6200 in the CANopen network.

TxPDO 1...4

A transmit-PDO transmits data from the CANopen gateway to a CANopen device.

Table 83: TxPDO 1...4

CAN ID	DLC	Data							
181-1FF	8	B1	B2	B3	B4	B5	B6	B7	B8
281-2FF	8	B9	B10	B11	B12	B13	B14	B15	B16
381-3FF	8	B17	B18	B19	B20	B21	B22	B23	B24
481-4FF	8	B25	B26	B27	B28	B29	B30	B31	B32

B1...B32: CAN telegram bytes as in the network input data, with the aid of samosPLAN5+ (see *Configuring the gateway output values (tab 1) [ch. 5.3, p. 40]*).

The gateway sends one or several TxPDOs when at least one of the following events occurs:

- At least one input or output byte has changed its value and the transmission type for the TxPDO that contains this byte has the value 255.
- At least one input or output byte has changed its value and the gateway contains a SYNC command and at least one TxPDO has transmission type 0.
- When the transmission type is n = 1 ... 240, n sync commands are required in order to send the TxPDO.
- The transmission type for a TxPDO is 254 or 255 and the event timer (SDO 1800,5 for TxPDO1) has a value of N > 0. In this case this TxPDO is sent every N ms.
- A TxPDO can also be called up with the aid of a remote transmission request (RTR). This requires a CAN telegram to the gateway that contains the CAN-ID of the desired TxPDOs with DLC = 0 and RTR = 1.

The operating state of the device must be "operational" for all transmission methods (see *Table "Network management for all NMT slaves" [ch. 10.4, p. 127]*).

RxPDO 1...4

A receive-PDO transmits data from a CANopen device to the CANopen gateway.

Table 84: RxPDO 1...4

CAN ID	DLC	Data							
201-1FF	8	B1	B2	B3	B4	B5	B6	B7	B8
301-2FF	8	B9	B10	B11	B12	B13	B14	B15	B16
401-3FF	8	B17	B18	B19	B20	B21	B22	B23	B24
501-4FF	8	B25	B26	B27	B28	B29	B30	B31	B32

B1...B32: CAN telegram bytes as for the gateway input data, with the aid of samosPLAN5+.

The transmission type 255 is preset for all RxPDOs. This means that the gateway immediately transmits the RxPDO data on to the main module. This setting cannot be changed.

10.8 SDO communication

SDOs are service data objects. They contain a wide spectrum of different data. This includes configuration as well as input and output data.

Contrary to PDO communication, the receipt of each SDO is answered at protocol level,

i.e. the receiving device sends a confirmation.

This CANopen PCS implementation supports the following protocols:

- SDO Download Expedited (write SDO)
- SDO Upload Expedited (read SDO)
- Upload SDO Segment Protocol (segmented reading of an SDO)

SDO Download Expedited (write SDO)

The client sends a request to server N. The 16-bit index an the sub-index for the SDO to be written form part of this message. In addition, the request contains 4 data bytes with the data to be written.

Table 85: Write SDO

CAN ID	DLC	Data							
600h + N	8	23h	SDO_L	SDO_H	SUB	Byte 1	Byte 2	Byte 3	Byte 4

SDO_L = SDO-Index, Low Byte

SDO_H = SDO-Index, High Byte

SUB = SDO-Subindex

The server then responds with a confirmation:

Table 86: SDO write confirmation

CAN ID	DLC	Data							
580h + N	8	60h	SDO_L	SDO_H	SUB	Byte 1	Byte 2	Byte 3	Byte 4

Byte 1 to 4 in the write confirmation contain zeros.

SDO Upload Expedited (read SDO)

The client requests the content of an SDO by submitting a request to server N. The 16-bit index and the sub-index for the SDO to be read form part of this message. Byte 1 to 4 in the read request contain zeros.

Table 87: Read SDO

CAN ID	DLC	Data							
600h + N	8	40h	SDO_L	SDO_H	SUB	Byte 1	Byte 2	Byte 3	Byte 4

The server responds with the following message. Bytes 1 to 4 contain the value of the requested object.

Table 88: SDO read confirmation

CAN ID	DLC	Data							
580h + N	8	42h	SDO_L	SDO_H	SUB	Byte 1	Byte 2	Byte 3	Byte 4

The CANopen data types UDINT and UINT

In order to transmit the data types UDINT or UINT, the data must be in Intel format. For example, the 32-bit value 12345678h in data bytes 5, 6, 7 and 8 must be transmitted in the following order: [5] = 78, [6] = 56, [7] = 34, [8] = 12.

NOTICE This also applies to the SDO index in data bytes 2 and 3, which is of the data type UINT. This means that the low byte is transmitted in data byte 2 and the high byte in data type 3.

Example: The following messages are required to read SDO 1003,1 of the CANopen device with device address 2. The data type of the data to be read is UDINT.

The client sends:

CAN ID	DLC	Data							
602h	8	40h	03h	10h	01h	00h	00h	00h	00h

The server responds:

CAN ID	DLC	Data							
582h	8	42h	03h	10h	01h	08h	00h	50h	02h

The combined response data result in the 32-bit word 02500008h.

10.9 SDO object directory

Each CANopen device manages its SDOs in an object directory. The complete

object directory is formally described in an EDS file. Many CANopen tools can ready this EDS file and therefore know the object characteristics of the CANopen device.

The following table shows all SDOs for the SP-CANopen gateway.

Table 89: Supported SDOs

SDO #	Type
1000	Device type
1001	Error register
1003	Error list (error history)
1005	COB ID SYNC
1008	Device name
1009	Hardware version
100A	Software version
100C	Guard Time
100D	Life Time Factor
1016	Consumer Heartbeat Time
1017	Producer Heartbeat Time
1018	Identification
1027	Module list
1400…1403	Communication parameter for RxPDO 1 … 4
1600…1603	Mapping parameter for RxPDO 1 … 4
1800…1803	Communication parameter for TxPDO 1 … 4
1A00…1A03	Mapping parameter for TxPDO 1 … 4
3100	Module status bits
3200	Config-CRC
3300	Module type code
6000	Process data input objects
6200	Process data output objects

You will find more detailed information about these SDOs in the CANopen standard draft DS 301 V4.02 (DSP 301 V4.1).

SDO 1001: Error register

The error register (SINT) contains an error bit indicating whether an error is present. If bit 0 is set to 1, a “generic error” has been detected.

SDO 1003: Error list (error history)

SDO 1003 is an array that contains the last 10 error codes that the gateway has reported with the aid of

emergency messages. Array index 0 contains the number of error codes recorded in SDO 1003.

A new error is recorded in index 1, while older errors will in this case be renumbered (incremented by 1). The array index can be overwritten with a 0 from the outside, thus completely deleting the array.

NOTICE

- Not all errors reported with the aid of emergency messages are recorded in SDO 1003, only the errors listed here:
  Error and state information for the modules [ch. 3.3.5, p. 19]
- The entries in SDO 1003 are in UDINT format and normally divided into 16 bits of error code and 16 bits of additional information. In the event of an emergency, the module state diagnosis (4 bytes) will be entered here.

SDO 1005: COB ID SYNC

SDO 1005 contains the COB-ID of the sync object. This value has been preset to 80h, but can be changed.

NOTICE

When you change the COB-ID of the sync object, please ensure that the new ID has not already been allocated to another communication object.

SDO 1008: Device name

SD0 1008 contains a device name (VISIBLE STRING).

NOTICE

This SDO cannot be read with a simple "SDO upload expedited". The "Upload SDO segment protocol" command (client command code ccs = 3) must be used instead, as described in the CANopen specifications DS 301.

SDO 1009: Hardware version

SDO 1009 contains the current hardware version of the device (VISIBLE STRING).

NOTICE

This SDO cannot be read with a simple "SDO upload expedited". The "Upload SDO segment protocol" command (client command code ccs = 3) must be used instead, as described in the CANopen specifications DS 301.

SDO 100A: Software version

SDO 100A contains the current software version of the device (VISIBLE STRING).

NOTICE

This SDO cannot be read with a simple "SDO upload expedited". The "Upload SDO segment protocol" command (client command code ccs = 3) must be used instead, as described in the CANopen specifications DS 301.

SDO 100C: Guard Time

The guard time (UINT) multiplied by the life time factor (SINT) results in the life guarding time.

Life Guarding Time [ms] = Guard Time [ms] × Life Time Factor

During the life guarding time, the master must send at least one node guarding

message to the slave. When the life guarding time is exceeded (life guarding error), the gateway reports a cable break error and sets all network process data to 0; the LED NS starts to flash red.

In the slave, life guarding is activated by the first node guarding message when the life guarding time has not been set to 0. When the guard time or the life time factor are set to 0 after activating life guarding, life guarding will be deactivated.

Also see: *Guarding protocols [ch. 10.10, p. 140]*.

SDO 100D: Life Time Factor

SDO 100D contains the Life Time Factor (SINT). See SDO 100C.

NOTICE

The Life Time Factor must either be = 0 (deactivated) or V 1.5.

SDO 1016: Consumer Heartbeat Time

The gateway is configured as a heartbeat consumer when SDO 1016 contains a value greater than

0 for the consumer heartbeat time. The consumer heartbeat time is given in ms.

The NMT master must send at least one node guarding message to the slave within this time. When the consumer heartbeat time is exceeded (life guarding error), the gateway reports a cable break error and sets all network process data to 0; the LED NS starts to flash red.

SDO 1017: Producer Heartbeat Time

The gateway can also act as a heartbeat producer, i.e. send a heartbeat signal.

This allows another device to detect whether the heartbeat producer (i.e. the gateway) is still functioning correctly.

The producer heartbeat time is given in ms. For internal processing it is rounded up

to the next higher multiple of 4. If the heartbeat time is set to 0, the heartbeat signal is deactivated.

The heartbeat signal consists of a cyclic CAN message with the identifier 700h + device address.

NOTICE

It is not possible to use heartbeat signals and life guarding messages simultaneously, as both functions make use of the same CAN identifier.

Also see: *Guarding protocols [ch. 10.10, p. 140]*

SDO 1018: Identification

This SDO contains basic information about the gateway.

Table 90: Content of SDO 1018

Subindex	Mapping	Format	Description
1	Manufacturer ID	UDINT	Unique manufacturer identification number (e.g. Wieland Electric)
2	Product description	UDINT	Device variant
3	Revision number	UDINT	Software version of the device
4	Serial number	UDINT	Serial number of the device

SDO 1027: Module list

The module list contains the module type and the module number (module ID) of all safe samosPRO COMPACT modules in the system.

Table 91: Content of SDO 1027

Subindex	Module	Format
1-3	Main module (SP-COPx)	SINT
4…15	Expansion modules (SP-SDIO or SP-SDI)	SINT

SDO 1400 ... 1403: Communication parameters for the RxPDOs

SDO 1400 to 1403 can be used to configure the communication parameters for RxPDOs 1 to 4, e.g. SDO 1400 defines the parameters for RxPDO 1, etc.

Table 92: Content of SDO 1400 ... 1403

Subindex	Mapping	Format	Description
1	COB ID	UDINT	CAN identifier for this PDO, write-protected
2	Receive mode	SINT	Fix 255 (asynchronous mode)

The receive mode (read/write) determines how the PDO is to be received. For RxPDOs, the receive mode has been set to 255 (asynchronous mode). In this mode, the data of a RxPDOs received are directly routed to the outputs.

NOTICE When the receive mode is set to a value other than 255, an error code is generated (abort code 0609 0030h, invalid parameter value).

SDO 1600 ... 1603: Mapping parameters for the RxPDOs

This SDO cannot be used, as mapping of the RxPDOs takes place with the aid of the samosPLAN5+.
Also see: Table *"Pre-sets for the content of the transmit process data objects (TxPDOs) of the SP-CANopen" [ch. 10.7, p. 130]*

SDO 1800 ... 1803: Communication parameters for the TxPDOs

SDO 1400 to 1403 can be used to configure the communication parameters for TxPDOs 1 to 4, e.g. SDO 1800 defines the parameters for TxPDO 1, etc.

Table 93: Content of SDO 1800 ... 1803

Subindex	Mapping	Format	Description
1	COB ID	UDINT	CAN identifier for this PDO, write-protected
2	Transmission type	SINT	Defines when the PDO is to be sent
5	Event timer	UINT	in ms

The transmission type for all TxPDOs to 255 (asynchronous mode, event-driven) has been preset.

The event timer contains the time in Ms for the cyclic transmission of the TxPDOs.

Transmission types for the TxPDOs

Table 94: Transmission types for the TxPDOs

TxPDO	Synchronous	Asynchronous	RTR
1, 2, 3, 4	0, 1…240	254, 255	253

NOTICE When the transmission type is set to an invalid value, an error code is generated (abort code 0030 0030h, invalid parameter value).

Synchronous: The synchronous transmission type 0 means that the TxPDO will be sent once a sync command has been received, but only when data have changed. The synchronous transmission types n = 1 … 240 mean that the TxPDO is sent once the nth sync command has been received.

Asynchronous, event-driven with change of state: The asynchronous transmission type 255 (without a configured event timer) means that the TxPDO is sent each time

at least one input bit contained in this PDO has changed.

Asynchronous, event-driven by timer: The asynchronous transmission type 254/255 (with a

configured event timer) means that the TxPDO is sent each time the

event timer has run out, e.g. a value of 500 signals to the event timer that the

gateway sends the relevant TxPDO every 500 ms.

RTR, on request: Transmission type 253 means that the TxPDO can be requested with the aid of an RTR (remote transmission request). This requires a CAN message to the gateway with DLC = 0, RTR = 1 and the COB-ID of the TxPDO. The gateway then responds with the requested TxPDO.

SDO 1A00 … 1A03: Mapping parameters for the TxPDOs

This SDO cannot be used, as mapping of the TxPDOs takes place with the aid of the samosPLAN5+.

Also see: Table *"Pre-sets for the content of the transmit process data objects (TxPDOs) of the SP-CANopen" [ch. 10.7, p. 130]*

SDO 3100: Module status bits

SDO 3100 contains the module status bits of the samosPRO COMPACT system (see *Error and state information for the modules [ch. 3.3.5, p. 19]*). Active bits are low (= "0").

Table 95: Content of SDO 3100

SDO array	Data set parameters	Module	Size
3100.1-3	Status of Module 0	CPU	UDINT
3100.4	Status of Module 1	Expansion	UDINT
…	…	…	…
3100.14	Status of Module 11	Expansion	UDINT
3100.15	Status of Module 12	Expansion	UDINT

NOTICE The positions of the modules are numbered in the samosPLAN5+ from 0 to 14. Thus the sub-index for SDO 3100 = Position + 3, with the first three sub-indices for the SP-COPx being used.

SDO 3100 can only be read.

SDO 3200: Config-CRC

SDO 3200 contains the system CRCs in UDINT format.

SDO 6000: Process data input objects

The 32 bytes of the process input data can be written into SDO array 6000. These are the same data as in RxPDO 1-4 (see *PDO Communication [ch. 10.7, p. 130]*). The mapping is as follows:

Table 96: Mapping table for SDO 6000 – RxPDO 1-4

SDO 6000	RxPDO
6000.1	RxPDO 1, Byte 1
…	…
6000.8	RxPDO 1, Byte 8
6000.9-16	RxPDO 2, Byte 1-8
6000.17-24	RxPDO 3, Byte 1-8
6000.25-32	RxPDO 4, Byte 1-8

SDO 6000 can only be written.

SDO 6200: Process data output objects

The 32 bytes of the process output data can be written into SDO array 6200. These are the same data as in TxPDO 1-4 (see *PDO Communication [ch. 10.7, p. 130]*). The mapping is as follows:

Table 97: Mapping table for SDO 6200 – TxPDO 1-4

SDO 6200	TxPDO
6200.1	TxPDO 1, Byte 1
…	…
6200.8	TxPDO 1, Byte 8
6200.9-16	TxPDO 2, Byte 1-8
6200.17-24	TxPDO 3, Byte 1-8
6200.25-32	TxPDO 4, Byte 1-8

SDO 6200 can only be read.

10.10 Guarding protocols

CANopen offers several possibilities for active monitoring of the correct function

of the field bus interface (e.g. cable break detection).

ATTENTION

Always use either node guarding or heartbeat!
Guarding is compulsory according to the CIA CANopen specifications DS 301. Please always active either node guarding or heartbeat. When no guarding has been configured, the samosPRO COMPACT system cannot detect an interruption of the CANopen communication, for example an interrupted network cable. In this case the input and output data of the CANopen gateway may "freeze".

Heartbeat

A heartbeat producer is a CANopen device that sends a cyclic heartbeat message. This makes it possible for all other CANopen devices to detect whether the heartbeat producer still functions correctly and what its current status is. Heartbeat messages are sent at regular intervals, i.e. the producer heartbeat

time, which may be configured with the aid of SDO 1017. The configured 16-bit value

is rounded up to the next higher multiple of 4 ms.

A heartbeat consumer is a CANopen device that expects a cyclic node guarding message within a certain time interval, i.e. the consumer heartbeat time, which can be configured with the aid of SDO 1016. If the heartbeat consumer does not receive a node guarding message within the configured consumer heartbeat time, it sends a life guarding emergency message and sets the process input data to 0. In addition, the gateway sends a "cable break" error message that can be processed by the main module.

Node guarding

Node guarding is carried out by a NMT master. This can be any CANopen device that can fulfill this function as a client. The NMT master sends a cyclic node guarding message to the device to be monitored, which must respond within a certain time, which is monitored by the NMT master. If the device to be monitored does not respond within the node guarding time, the NMT master treats this as a malfunction of the device and takes the corresponding actions.

Life Guarding

Life guarding is carried out by the gateway itself. In the gateway, the life guarding time is calculated from the values of SDO 100C (guard time) and SDO 100D (life time factor). If the gateway does not receive a node guarding message from an NMT master once within this life guarding time, the gateway sends an internal "cable break" error message, which can be processed by the main module, and the LED NS starts to flash red.

NOTICE

- The gateway can detect a cable break when life guarding has been activated, i.e. when both SDO 100C and SDO 100D have a value not equal to 0. In this case life guarding starts as soon as the first node guarding request is received from an NMT master and ends when the master sends the command "Reset communication".
- Alternatively cable break detection is possible when the gateway has been configured as a heartbeat consumer. In this case the cable break detection is carried out by the gateway itself.
- Heartbeat (producer) works without node guarding. In this case gateway cannot detect a cable break on the field bus.
- Heartbeat and node guarding / life guarding cannot be simultaneously used.
- If the configuration has been changed in such a way that life guarding is deactivated or activated, the entire samosPRO COMPACT System must be restarted, so that the CANopen network communication can again be correctly established.

The following table provides an overview of the supported guarding protocols, depending on the configuration of SDO 1016 and SDO 1017 (heartbeat), SDO 100C (guard time) and SDO 100D (life time factor).

Table 98: Overview and comparison of the guarding protocols

SDO 1016	SDO 1017	SDO 100C × 100D	Heartbeat gateway	Life Guarding Gateway	Node guarding NMT master
0	0	0	Not permitted: Always make use of either node guarding or heartbeat!		
0	0	> 0	Deactivated	**Cable break detection**	**Required**
> 0	0	0	**Cyclic heartbeat (consumer)**	**Cable break detection**	**Possible for other slaves**
0	> 0	0	**Cyclic heartbeat (producer)**	Not possible	Not possible, but guarding as a heartbeat consumer is possible
> 0	> 0	0	**Cyclic heartbeat (producer und consumer)**	Cable break detection	Not possible
> 0	> 0	> 0	**Not permitted**		

NOTICE It does not make sense to use heartbeat and life guarding simultaneously.

10.11 Error objects

The SP-CANopen reports CAN-specific errors (e.g. initialization errors, cable breaks, CAN communication errors) to the main module as SBUS+ errors.

Emergency object

The emergency producer (CANopen gateway) sends the emergency object to the emergency consumer (any CANopen device, usually the controls) when CAN-specific errors occur or an error state occurs.

The emergency object is sent as described in DS 301 (CANopen specifications) in accordance with the following table:

Table 99: Emergency states and transitions

Emergency state Before	Transition	Module-specific alarms	Emergency state After
Error-free	1	Incoming error	Error occurred
Error occurred	2	Error removed, other errors pending	Error occurred
Error occurred	3	Incoming error, other errors pending	Error occurred
Error occurred	4	All errors removed	Error-free

The gateway is in one of two possible emergency states, either *error-free* or *errors detected*. Emergency objects are sent, depending on the transitions between these two emergency states. The error code in the emergency object shows the emergency state in which the gateway currently is (also see table below).

Overview of error objects

Table 100: CAN-specific errors

Error	Error code SBUS+	Error type	Emergency error code Error register M1...M5	Error history SDO 1003	Results/possible remedy
CAN data overflow CAN control overflow in Rx Fifo	0x4501	Warning	0x8110 0x11 1, 0, 0, 0, 0	–	• CAN messages have been lost. • Limited band width. • Check the CAN settings, increase the baud rate, reduce the number of participants or the data volume.
CAN-error-passive CAN control takes place in an error-passive state	0x4503	Warning	0x8120 0x11 0, 0, 0, 0, 0	–	The gateway is only sending recessive bits, i.e. it is invalidating its own messages. The cause is either a hardware fault on the gateway or an external malfunction of the data transmission. • Check the cabling.

Error	Error code SBUS+	Error type	Emergency error code Error register M1…M5	Error history SDO 1003	Results/possible remedy
CAN bus off The CAN controls are in the bus off state	0x4504	Warning	–	–	Major transmission error. The CAN controls have separated the connection to the bus. Possible hardware defect. • Switch the samosPRO COMPACT system off and on again.
CAN-Tx-Fifo overflow The CAN controls have no transmission resources	0x4506	Warning	0x8110 0x11 2, 0, 0, 0, 0	–	CAN messages that were to be sent from the gateway have been lost. The number of events for which the gateway is to send CAN messages is too high for the set baud rate. • Increase the baud rate or change the configuration of the gateway.
CAN initialization failed. The CAN controls could not be initialized	0xC507	Critical	–	–	The CAN controls or the transceiver may be defective. • Replace the SP-CANopen with a new device.
CANopen Life Guarding CANopen Life Guarding has found a cable break	0x4508	Warning	0x8130 0x11 0, 0, 0, 0, 0	–	The gateway has generated a life guarding error message: Either an error has occurred on the node guarding or the heartbeat NMT master or the CAN cable has been interrupted. • Check the CANopen master. • Check the cabling.

10.12 CANopen diagnostic examples

Reading the current error from SDO 3100:

PLC requests:

CAN-ID	DLC	DATA							
60C	8	40	00	31	04	00	00	00	00

60C: Identifier (600 + C)

8: Data length code: This is followed by 8 bytes

40: Expedited upload requirement

00 31: Index 3100

04: Subindex: Module to Position 1
(see table *"Content of SDO 3100" [ch. 10.9, p. 138]*)

Gateway response:

CAN-ID	DLC	DATA							
58C	8	42	00	31	04	BF	FF	FF	FB

58C: Identifier (580 + C)

8: Data length code: This is followed by 8 bytes

42: Upload response, size of data set is not shown

00 31: Index 3100

04: Subindex: Module to Position 1
(see table *"Content of SDO 3100" [ch. 10.9, p. 138]*)

FB: Error byte M5, Bit 2 = 0: external error

BF: Error byte M2, Bit 30 = 0. Error: Short-circuit after high at output 4

Reading the current error from SDO 3100:

PLC requests:

CAN-ID	DLC	DATA							
60C	8	40	00	31	0F	00	00	00	00

60C: Identifier (600 + C)

8: Data length code: This is followed by 8 bytes

40: Expedited upload requirement

00 31: Index 3100

0F: Subindex 0F = Module in Position 12 (module position = subindex - 1, also see table *"Content of SDO 3100" [ch. 10.9, p. 138]*)

Gateway response:

CAN-ID	DLC	DATA							
58C	8	42	00	31	0F	FF	FF	FE	FB

58C: Identifier (580 + C)

8: Data length code: This is followed by 8 bytes

42: Upload response, size of data set is not shown

00 31: Index 3100

0F: Module to Position 12
(see table *"Content of SDO 3100" [ch. 10.9, p. 138]*)

FB: Error byte M5, Bit 2 = 0: external error

FE: Error byte M4, bit 0 = 0: two-channel evaluation of inputs 1–2: Error detected
(see *Error and state information for the modules [ch. 3.3.5, p. 19]*)

Reading of error from the error history in SDO 1003:

PLC requests:

CAN-ID	DLC	DATA							
60C	8	40	03	10	01	00	00	00	00

60C: Identifier (600 + C)

8: Data length code: This is followed by 8 bytes

40: Expedited upload requirement

03 10: Index 1003

01: Sub-index: last error

Gateway response:

CAN-ID	DLC	DATA							
58C	8	42	03	10	01	00	00	01	00

58C: Identifier (580 + C)

8: Data length code: This is followed by 8 bytes

42: Upload response, size of data set is not shown

03 10: Index 1003

01: Sub-index: last error

01: Module status bit 8 (bit 0 of byte M4) = 0: two-channel evaluation of inputs 1–2: error detected

10.13 Diagnosis and troubleshooting

Information about the diagnosis of the samosPRO COMPACT system may be found in the "samosPLAN5+ Software" manual (BA000968).

Table 101: Error removal for SP-CANopen

Error		Possible cause	Possible remedy
Key: ○ LED off / ✳ LED flashes / ● LED lights up			
The SP-CANopen is not providing any data.		• Configuration required, node guarding or heartbeat message was not sent. • The configuration has not yet been fully transmitted.	• Configure the SP-CANopen and transfer the configuration to the system. • Wait until the configuration has been fully transferred.
LED PWR	● Green		
LED NS	○ Off		
MS LED	✳ Red (1 Hz)		
The SP-CANopen is not providing any data.		The configuration has not yet been fully transmitted.	Wait until the configuration has been fully transferred.
LED PWR	● Green		
LED NS	● Green		
MS LED	✳ Red (1 Hz)		
The SP-CANopen is not providing any data.		No PDO transfer since switch-on.	• Start the PDO transfer. • Transfer the PDO via SDO 6000 or SDO 6200.
LED PWR	● Green		
LED NS	● Green		
MS LED	✳ / ✳ Red / green		
The SP-CANopen is not providing any data.		• No PDO transfer since switch-on. • Wrong baud rate (CAN transceiver possibly in error passive). • Wrong node ID or CANopen address. • The CAN cable was interrupted.	• Start the PDO transfer. • Transfer the PDO via SDO 6000 or SDO 6200. • Check and correct the baud rate. • Check and correct the address. • Check the CANopen cabling.
LED PWR	● Green		
LED NS	✳ Green		
MS LED	✳ / ✳ Red / green		
The SP-CANopen is not providing any PDO data		• SP-CANopen is in an idle state. • Node guarding or heartbeat messages are sent. • samosPRO COMPACT configuration has not been verified and the main module has been stopped	• The CPU/application has stopped. • Start the CPU (change to Run mode). • Verify the configuration with the samosPLAN5+ and start the CPU module.
LED PWR	● Green		
LED NS	○ Off /I Red /I Green		
MS LED	✳ Green (1 Hz)		

Error		Possible cause	Possible remedy
The SP-CANopen is not providing any PDO data		Supply voltage too low.	Check the power supply.
LED PWR	Green		
LED NS	Green		
MS LED	Off		
The SP-CANopen is not providing any data.		Brief drop in power supply.	• Check the power supply. • Reset the samosPRO COMPACT system
LED PWR	Red		
LED NS	Red		
MS LED	Red		
The SP-CANopen is not providing any data.		• Wrong node ID or CANopen address. • Wrong baud rate (CAN transceiver possibly in error passive), SP-CANopen is in idle state.	• Check and correct the address. • Check and correct the baud rate.
LED PWR	Green		
LED NS	Green (1 Hz)		
MS LED	Green (1 Hz)		
The SP-CANopen is not providing any data.		• Wrong baud rate and the transceiver of the SPCANopen is in bus-off state (hardware problem at the physical CAN level). • The CAN cable was interrupted.	• Check and correct the baud rate. • Check the CANopen cabling. • Reset the samosPRO COMPACT system.
LED PWR	Green		
LED NS	Red		
MS LED	Red / green		
The SP-CANopen is not providing any data.		• CANopen master is in the stop or pre-operational state • Another slave could not be initialized during initialization of the bus system. • CANopen state of the SP-CANopen is pre-operational. Wrong node ID or CANopen address.	• Set the CANopen master to the run state (CANopen state operational). • Check whether all slaves on the bus have been switched on. • Check the CANopen cabling. • Check whether the CAN master starts automatically. • Check and correct the CANopen address.
LED PWR	Green		
LED NS	Green (1 Hz)		
MS LED	Green		
The SP-CANopen is not providing any data.		• The transceiver of the SP-CANopen is in error passive state. • The CAN cable was interrupted.	• Check the CANopen cabling. • Check the diagnostic messages with the aid of the samosPLAN5+. • Reset the samosPRO COMPACT system.
LED PWR	Green		
LED NS	Red		
MS LED	Green		

Error		Possible cause	Possible remedy
The SP-CANopen is not providing any data.		• Node guarding or heartbeat consumer failure • The guarding configuration was changed.	• Check the CANopen cabling. • Check the life guarding time (life time factor V 1). • Check the heartbeat consumer time (should be V 1.5 × heartbeat producer time). • Check the diagnostic messages with the aid of the samosPLAN5+. • Reset the samosPRO COMPACT system.
LED PWR	Green		
LED NS	Red (1 Hz)		
MS LED	/ Red / green		
The SP-CANopen is in the critical error state		• Internal equipment error on the SP-CANopen. • The module version of the CPU does not support any samosPRO COMPACT gateways.	• Switch the samosPRO COMPACT system power supply off and on again. • Check the diagnostic messages with the aid of the samosPLAN5+. • Use a CPU with the required module version (see Proper Use). • If the error persists, replace the gateway.
LED PWR	Green		
LED NS	Red		
MS LED	Red (2 Hz)		
The SP-CANopen / samosPRO COMPACT system is in the **critical error** state		• The SP-CANopen is not properly connected to the other samosPRO COMPACT modules. • The module connection plug is dirty or damaged. • Another samosPRO COMPACT module has an internal critical error.	• Correctly plug in the SP-CANopen. • Clean the connection plug and socket. • Switch on the power supply • once again. • Check the other samosPRO COMPACT modules.
LED PWR	Red		
LED NS	Off		
MS LED	Red		

11 EtherCAT Gateway

The samosPRO EtherCAT gateway can only be used in combination with the samosPRO COMPACT module from module version C-xx. The EtherCAT gateway offers samosPLAN5+ software from version 1.3.0 onwards.

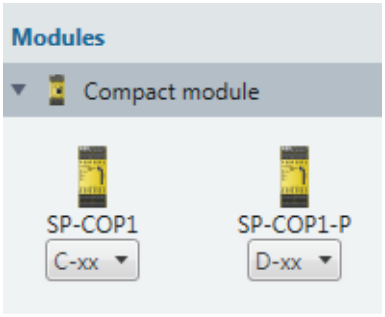


Illustration 54: Setting the module version in the 'Hardware' view

The module version defines the current version of the hardware and software and can also be read on the side of the housing. Later versions have different letters as the first letter of the module version in alphabetical order (e.g. the module version **D-xx** would be a more recent version).

Illustration 55: Side label on the samosPRO COMPACT module

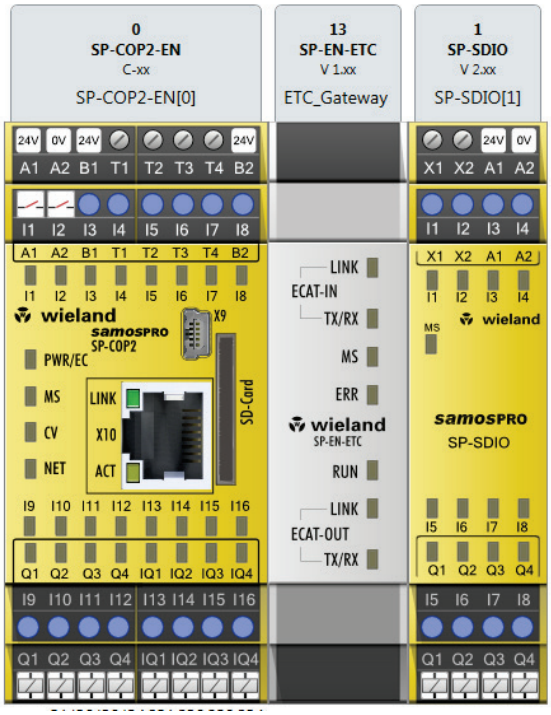


Illustration 56: Configuration example: SP-COP2-EN (0), SP-EN-ETC(13), SP-SDIO(1)

11.1 Interfaces and operation

Operation and display elements

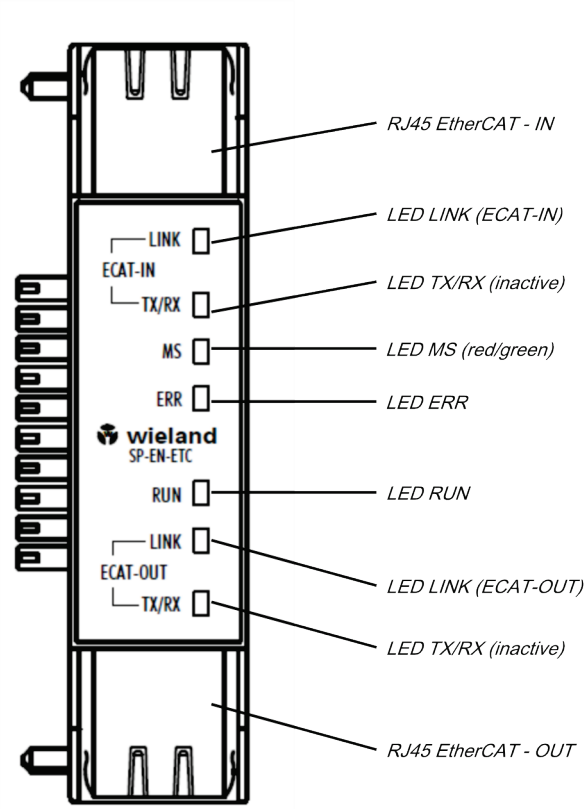


Illustration 57: Operation and display elements of the SP-EN-ETC

Table 102: Meaning of the status LEDs on the SP-EN-ETC

LED		Meaning
Key: ○ LED off / ✳ LED flashes / ● LED lights up		
ECAT-IN		
LINK	○ Off	No EtherCAT device connected, no connection.
	● Green	EtherCAT device connected.
	✳ Green	Communication with connected EtherCAT device
TX/RX	○ Off	Not used
MS	○ Off	No voltage supply / No connection to the head-end station
	● Green	On: samosPRO in operation
	✳ Green	Flashing 1 Hz: samosPRO stopped
	✳ / ✳ Red / green	Alternate flashing: Run but the gateway has an error (e.g. no EtherCAT connection)

LED		Meaning
	 Red	Flashing 1 Hz: Configuration required or is taking place right now
	 Red	On: Critical errors
ERR	 Off	**No error:** The EtherCAT communication of the device is in operation
	 Red	**Double flash** **Application watchdog timeout:** An application watchdog timeout occurred (Example: Sync Manager watchdog timeout)
	 Red	**Single flash** **Unrequested status change:** The slave device application has autonomously changed the EtherCAT status: The "Change" parameter in the ALStatus register is 0x01:change/error.
	 Red	**Blink** **Invalid configuration:** General configuration error (Example: The configuration has not yet been fully transmitted.)
	 Red	**On** **Watchdog timeout:** A watchdog timeout has occurred. (Example: The application controller is no longer responding)
RUN	 Off	**Off** **"INIT":** The device is in the INIT state.
	 Green	**On** **"OPERATIONAL"**
	 Green	**Blink** **"PRE-OPERATIONAL"**
	 Green	**Single flash** **"SAFE-OPERATIONAL"**
ECAT-OUT		
LINK	 Off	No EtherCAT device connected, no connection
	 Green	**On** EtherCAT device is connected
	 Green	**Blink** The device sends/receives Ethernet frames
TX/RX	 Off	This LED is not used

Table 103: Information about the light behavior of the EtherCAT status LEDs

LED states	Description
On	The indicator is constantly on.
Off	The indicator does not come on.

LED states	Description
Blink	The indicator is switched on or off in phases at a frequency of 2.5 Hz.
Single flash	The indicator shows one short flash (200 ms) followed by a longer off phase (1000 ms).
Double flash	The indicator shows a sequence of two short flashes (200 ms each) interrupted by a short off phase (200 ms). The sequence is finished by a long off phase (1000 ms).

11.2 EtherCAT basics

General information

Field buses have been established in automation engineering for many years. Since on the one hand there is demand for ever higher speeds, but on the other hand the technical limits have already been reached with this technology, new solutions must be sought.

The Ethernet known from the office world, with its available-everywhere 100Mbit/s, is very fast. The type of cabling used there and the rules governing access rights mean that this Ethernet is not real-time capable. This effect has been rectified with EtherCAT.

EtherCAT

For EtherCAT: EtherCAT is a registered trademark and patented technology, licensed by Beckhoff Automation GmbH, Germany.

EtherCAT stands for Ethernet for Controller and Automation Technology. It was originally developed by Beckhoff Automation GmbH and is now supported and further developed by the EtherCAT Technology Group (ETG). The ETG is the world's largest international users and manufacturers association for industrial Ethernet with around 1450 member firms (as at October 2010).

EtherCAT is an open Ethernet-based fieldbus system that is standardized in the IEC. As an open fieldbus system, EtherCAT satisfies the user profile for the area of industrial real-time systems.

Unlike traditional Ethernet communications, in EtherCAT the I/O data are exchanged at 100MBit/s in full duplex mode, while the telegram passes through the coupler. Since in this way a telegram reaches lots of devices in the transmit and receive direction, EtherCAT has a useful data rate of over 90%.

The EtherCAT protocol, optimized for process data, is transported directly in the Ethernet telegram. In turn, this can consists of several sub-telegrams, each serving one memory area of the process image.

Transmission medium

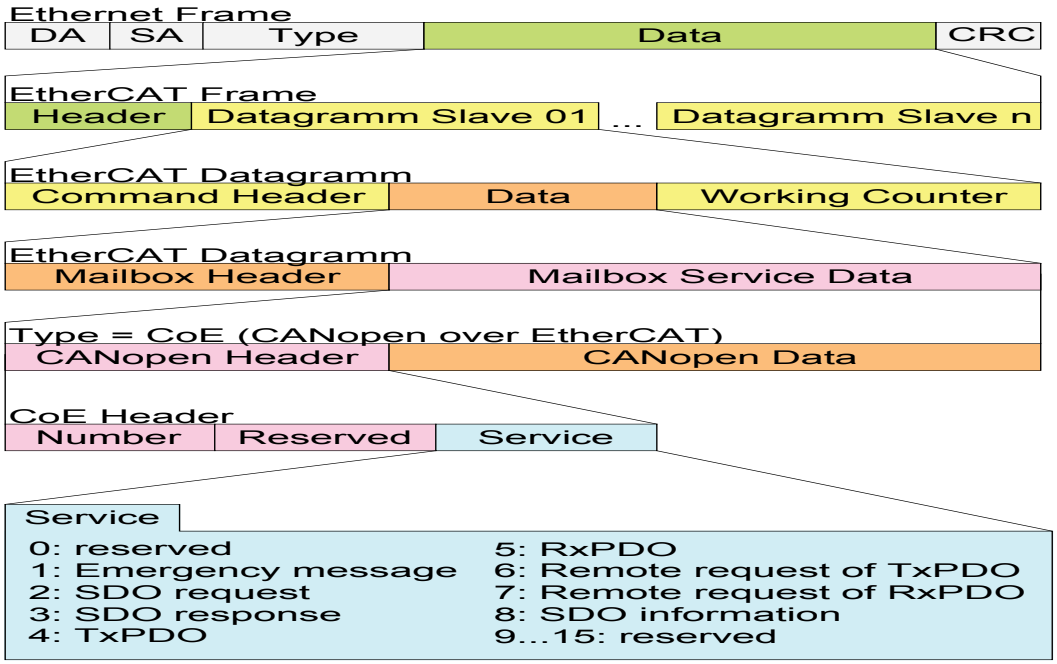
EtherCAT uses Ethernet as the transmission medium. Standard CAT5 cable is used. Cable lengths of up to 100m between 2 devices are possible.

Only EtherCAT components may be used in an EtherCAT network. To implement topologies deviating from the linear structure corresponding EtherCAT components are required that support this. It is not possible to use hubs.

Communication principle

In EtherCAT the master sends a telegram to the first device. This extracts the data intended for it from the data flow, inserts its response data into the telegram and sends the telegram on to the next device. The next device processes the telegram in the same way.

If the telegram has reached the last device, this recognizes that no more devices are connected and sends the telegram back to the master. In this way the telegram is sent via the other pair of wires through all devices to the master (full duplex). The connection sequence and the use of full-duplex technology means EtherCAT is a logical ring.



DA	Destination address	SA	Source address
CRC	Checksum	Type	Ether type (example: the entry 0x88A4 means EtherCAT protocol.)

Components

The components of the CoE interface are listed below:

EtherCAT State Machine

The EtherCAT State Machine controls the state of the EtherCAT coupler.

Object directory

The object directory lists all parameter, diagnostic, process or other data which can be read or described via EtherCAT. The SDO information service provides access to the object directory.

Process data

The EtherCAT data link layer is optimized for the fast transfer of process data. This determines how the process data of the device is assigned to the EtherCAT process data and how the application on the device is synchronized to the EtherCAT cycle.

The assignment of the process data (mapping) is done via the PDO Mapping and the SyncManager PDO Assign objects. These describe which objects from the object directory are transferred as process data with EtherCAT. The SyncManager Communication objects determine the cycle time with which the associated process data are transferred via EtherCAT and in what form it is synchronized for transmission.

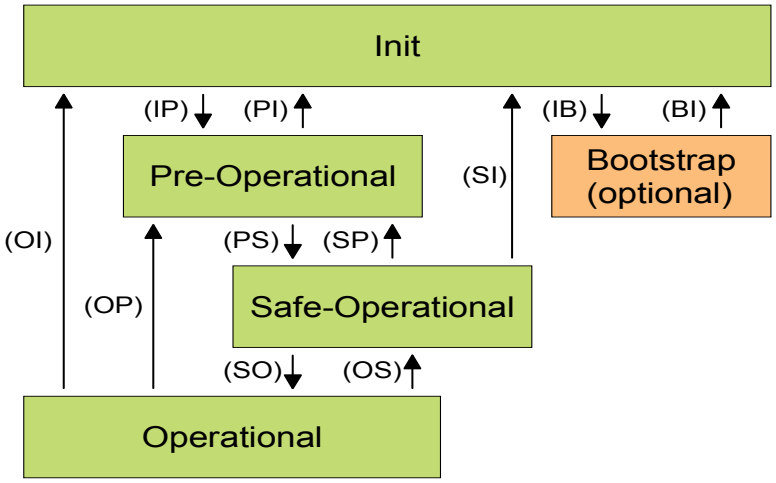
ESI file: Wieland SP EN ETC V1.1.xml

You will receive an ESI file from Wieland for the EtherCAT gateway. This file is located either on the enclosed disk or in the download area of www.wieland-electric.com. Install the ESI files in your PLC software configuration tool. Further details on installation of the ESI files can be found in the PLC manual.

11.3 EtherCAT state machine

States

A state machine is implemented in every EtherCAT coupler. For each state it is defined which communication services are active via EtherCAT. The state machine is controlled by the EtherCAT master.



IP	Start mailbox communication	PI	Stop mailbox communication
PS	Start input update	SP	Stop input update
SO	Start output update	OS	Stop output update
OP	Stop input update, stop output update	SI	Stop input update, stop mailbox communication
OI	Stop output update, stop input update Stop mailbox communication	IB	Start mailbox for firmware update in bootstrap mode (not implemented)
BI	Restart/stop mailbox		

INIT

After being switched on, the EtherCAT coupler is in the "Init" state. In this state neither mailbox nor process data communication are possible. The EtherCAT master initializes the SyncManager channels 0 and 1 for mailbox communication.

Pre-Operational (Pre-OP)

In the transition from **Init** to **Pre-Op**, the EtherCAT checks whether the mailbox was initialized correctly.

In the **Pre-Op** state mailbox communication is possible but not process data communication. Furthermore, in this state the settings for the transmission of process data and module-specific parameters are transmitted where they deviate from the standard settings.

Safe-Operational (Safe-OP)

In the transition from Pre-Op to Safe-Op the EtherCAT coupler checks whether the channels for process data communication are correct. Before it acknowledges the state change, the EtherCAT gateway copies current output data into the corresponding DP RAM areas of the EtherCAT gateway controller. In the Safe-Op state mailbox and process data communication are possible. Here the output data are updated cyclically while the input data are set to zero.

Operational (Op)

In the "Op" state the EtherCAT gateway copies the data in the RX-PDO onto its input data set 1. The output data set 1 is copied by the gateway into the TX-PDO and sent to the EtherCAT master.

Bootstrap optional (Boot)

not implemented

11.4 Bus topology and cabling

EtherCAT uses Ethernet as the transmission medium. Standard CAT5 cable is used. Cable lengths of up to 100 m between 2 devices are possible.

Only EtherCAT components may be used in an EtherCAT network. To implement topologies deviating from the linear structure corresponding EtherCAT components are required that support this. It is not possible to use hubs.

An EtherCAT network always consists of a master and any number of EtherCAT slaves (gateways or couplers). Each EtherCAT slave has an RJ45 socket **IN** and **OUT**. The incoming EtherCAT cable from the direction of the master should be plugged into the socket labeled **IN**. The RJ45 socket **ECAT-OUT** is used to connect further EtherCAT devices in the same strand in order to create so-called "daisy chains". In the last device the **OUT** socket remains free.

EtherCAT RJ45 bus interface

Note: The device supports the Auto Crossover function.

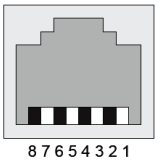
Pin	Signal	Setup
1	TX+	 8 7 6 5 4 3 2 1
2	TX-	
3	RX+	
4	Term 1	
5	Term 1	
6	RX-	
7	Term 2	
8	Term 2	
Housing	Screen	

Table 104: Ethernet connection data

Pin	Signal
Medium	2 x 2 pair twisted copper cable, CAT5 (100 MBit/s)
Cable length	max. 100 m
Transfer rate	100 MBit/s

Important information

- Use of hubs:
  Hubs are **not permitted** in EtherCAT networks.
- Use of switches:
  Switches in EtherCAT networks are only permitted between EtherCAT master and the first EtherCAT slave (100 Mbit/s, full duplex). Wieland Electric GmbH offers its own switches under the product family name "Ethernet Switch".
- Terminator:
  If the gateway is the last device, the EtherCAT topology does not require a terminator.
- Recommendation

  Take appropriate measures to protect the data cables and connectors against high mechanical load. We recommend a fixed installation in conjunction with tension relief.

11.5 Data transferred into the network

Available data

The samosPRO EtherCAT gateway can provide the following data:

- Operating data
  - **Logic results of** samosPRO COMPACT (see *Routing table [ch. 5.1.2, p. 33]*)
  - **Input values** (HIGH/LOW) for all samosPRO COMPACT input expansion modules in the system
  - **Output values** (HIGH/LOW) for all samosPRO COMPACT input/output expansion modules (see *Module state / input and output values [ch. 3.3.1, p. 17]*)
  - **Output data** from another network, i.e. data received from a second gateway in the samosPRO COMPACT system (see *Transmission of data from a second network [ch. 3.3.3, p. 18]*)
- Diagnostics
  - **Test values** (CRCs) (see *Configuration test values (CRCs) [ch. 3.3.4, p. 18]*)
  - **Error and state information** for all modules except SA-OR-S2 and SA-OR-S1 (see *Error and state information for modules [ch. 3.3.5, p. 19]*)

Data sets

The physical samosPRO COMPACT modules are not presented as typical hardware modules in the network. Instead, the data provided by the samosPRO COMPACT system have been arranged in three input data sets.

11.5.1 Data set 1

Data set 1 (50 bytes) contains the process data. It can be compiled with the aid of samosPLAN5+. In the form in which it is delivered, the content of data set 1 is preconfigured; it can be freely modified.

Note: **Not allocated** means that the byte value is equal to 0x00. However, the user can freely assign these bytes.

Table 105: Data set: Output data set 1 samosPRO COMPACT to --> SP-EN-ETC

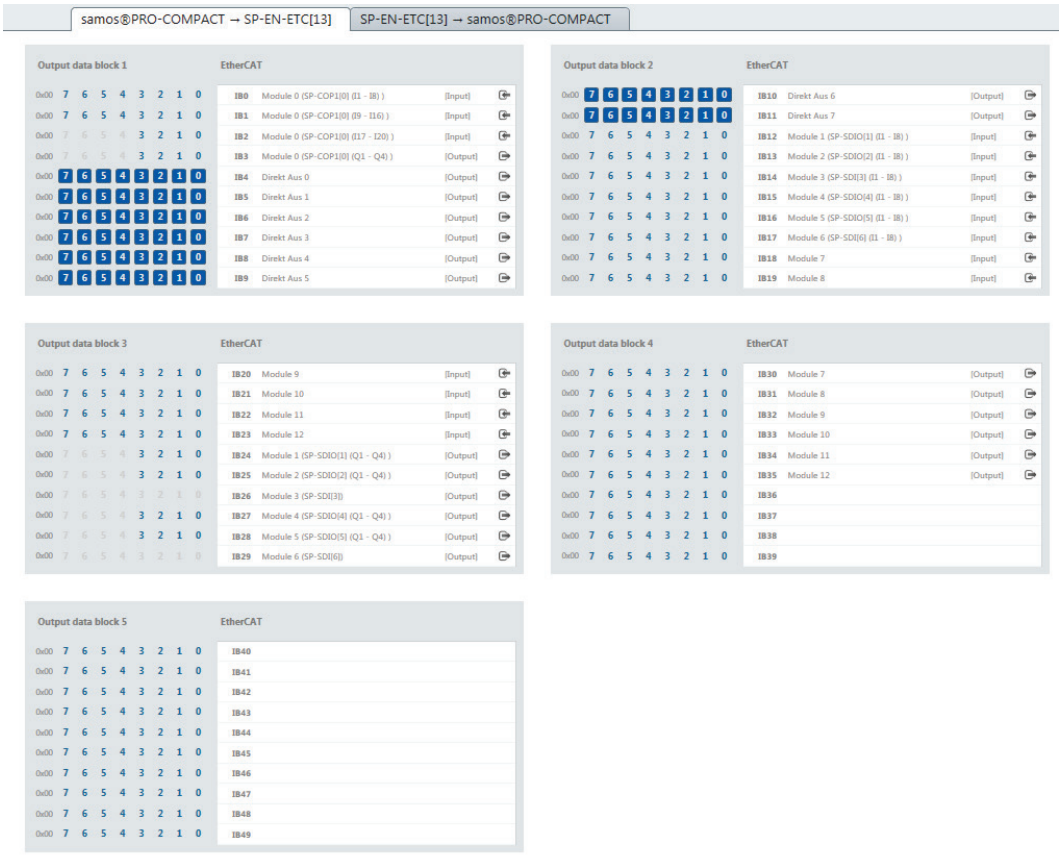
Output data block 1		Output data block 2	
Byte 0	Input values for Module 0 (I1..I8)	Byte 10	Not allocated
Byte 1	Input values for Module 0 (I9..I16)	Byte 11	Not allocated
Byte 2	Input values for Module 0 (IQ1..IQ4)	Byte 12	Input values for Module 1
Byte 3	Output values for Module 0 (Q1..Q4, IQ1..IQ4)	Byte 13	Input values for Module 2
Byte 4	Not allocated	Byte 14	Input values for Module 3
Byte 5	Not allocated	Byte 15	Input values for Module 4
Byte 6	Not allocated	Byte 16	Input values for Module 5
Byte 7	Not allocated	Byte 17	Input values for Module 6
Byte 8	Not allocated	Byte 18	Input values for Module 7
Byte 9	Not allocated	Byte 19	Input values for Module 8

Output data block 3		Output data block 4	
Byte 20	Input values for Module 9	Byte 30	Output values for Module 7
Byte 21	Input values for Module 10	Byte 31	Output values for Module 8
Byte 22	Input values for Module 11	Byte 32	Output values for Module 9
Byte 23	Input values for Module 12	Byte 33	Output values for Module 10
Byte 24	Output values for Module 1	Byte 34	Output values for Module 11
Byte 25	Output values for Module 2	Byte 35	Output values for Module 12
Byte 26	Output values for Module 3	Byte 36	Not allocated
Byte 27	Output values for Module 4	Byte 37	Not allocated
Byte 28	Output values for Module 5	Byte 38	Not allocated
Byte 29	Output values for Module 6	Byte 39	Not allocated

Output data block 5	
Byte 40	Not allocated
Byte 41	Not allocated
Byte 42	Not allocated
Byte 43	Not allocated
Byte 44	Not allocated
Byte 45	Not allocated
Byte 46	Not allocated
Byte 47	Not allocated
Byte 48	Not allocated
Byte 49	Not allocated
Total length	**50 bytes**

Tag names pre-assigned in samosPLAN5+ for the EtherCAT gateway

The data set 1 is divided into five input data blocks for clarity, whereby data blocks 1 to 5 each contain 10 bytes.



Direct gateway output values

It is possible to write values directly from the logic editor to the gateway. These valuesare freely programmable and are transferred to the EtherCAT network in the Transmit PDO. Four bytes have been reserved for this purpose in the basic settings for data set 1; however, up to the total number of 50 bytes of data set 1 may be configured as direct gateway output values. You can obtain additional information at: *Direct gateway output values [ch. 3.3.1, p. 17]*

Module state / input and output values

The samosPRO COMPACT gateway can transmit the input and output states of all samosPRO COMPACT modules connected to the samosPRO COMPACT system to the network. Data set 3 contains a non-modifiable configuration. Moreover, data set 1 can be adapted to contain up to 4 bytes of collective state information. Only the input and output values for data set 1 have been predefined and these can be freely adapted. You will find more detailed information in the section on the relevant gateway, as well as in the following section: *Configuration of gateways with samosPLAN5+ [ch. 5, p. 31]*

Module state

The samosPRO COMPACT gateway can transfer the state of the linked modules to the network. A total of 4 bytes are available for this purpose.

Table 106: Module state

Module state	Size	Meaning	Assignment
Input data state	2 bytes	One sum bit per module for the state of the module inputs 0 = error 1 = no error	Bit 0 = SP-COPx Bit 1 = 1. Module Bit 2 = 2. Module … Bit 12 = 12. Module Bit 13 = 1. Gateway Bit 14 = 2. Gateway Bit 15 = reserved
Output data state	2 bytes	One sum bit per module for the state of the module outputs 0 = error 1 = no error	

You will find information about the meaning of the state bits at: "samosPLAN5+ Software" manual (BA000968), chapter "Status bits for type SP-COP modules (reference)"

- **Input values for I/O modules**

  1 byte for data set 1 is available for every expansion module. The input values show the state of the preliminary evaluation of the I/O module. This corresponds to the state of the element in the main module logic. The level at the associated terminal cannot be clearly detected from this, as the data may be set to low, irrespectively of the level at the input terminal, by means of the cross-connection detection or two-channel evaluation (e.g. I1-18).

  When two-channel input elements have been configured for an I/O module, only the lower-value bit represents the pre-evaluation state of the corresponding element (e.g. bit 0 for I1 and I2, bit 2 for I3 and I4, bit 4 for I5 and I6, bit 6 for I7 and I8). The higher-value bit (bit 1, 3, 5 and 7) is used as follows in this case:

  0 = error 1 = no error

Table 107: Module state (input data status, byte 1)

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Module 7	Module 6	Module 5	Module 4	Module 3	Module 2	Module 1	SP-COPx

Table 108: Module state (input data status, byte 2)

Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8
Reserved	Gateway 2	Gateway 1	Module 12	Module 11	Module 10	Module 9	Module 8

- **Output values for I/O modules**

  1 byte for data set 1 is available for every module with outputs. The output values indicate the state of the control information from the logic of the main module for the relevant element of the I/O module. The level of the associated terminals cannot be clearly detected from this, as the output may be switched off via the cross-connection detection or the overload connection function.

  When two-channel output elements have been configured for an I/O module, only the lower-value bit represents the control information (e.g. bit 0 for Q1 and Q2, bit 2 for Q3 and Q4, bit 4 for Q5 and Q6, bit 6 for Q7 and Q8). The higher-value bit (bit 1, 3, 5 and 7) is not used as follows in this case (low):

Table 109: Module state (output data status, byte 1)

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Module 7	Module 6	Module 5	Module 4	Module 3	Module 2	Module 1	SP-COPx

Table 110: Module state (output data status, byte 2)

Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8
Reserved	Gateway 2	Gateway 1	Module 12	Module 11	Module 10	Module 9	Module 8

Transmission of data from a second network

If your samosPRO COMPACT system contains two gateways, it is possible to forward information which the first gateway receives from a network (e.g. from an EtherCAT PLC) via the second gateway to a second network (e.g. to a PROFIBUS master) and vice versa.

Expert setting: Allocating bytes to other addresses

samosPLAN5+ has pre-assigned the addresses according to a default. You can manually change this address allocation by moving any number of bytes.

In our example, we have shifted **byte 1** to **byte 23** in output data block **1**.

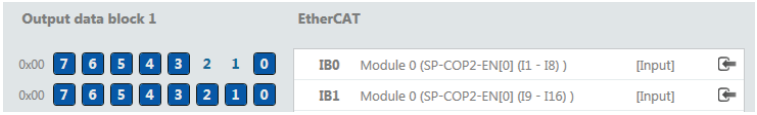


Illustration 58: Initial situation

➡ Ensure that the desired address (**byte 23** in our example) has not been allocated. In our example, **byte 23** has already been allocated to **module 12**. We therefore first have to delete **byte 23**.

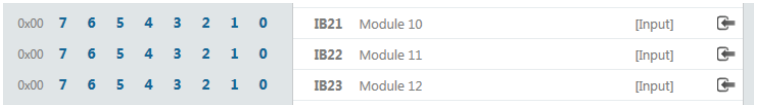


Illustration 59: Byte 23

➡ Delete the byte you wish to reallocated (**byte 1** in our example). To do this, click on the byte in the work area and click on the **Delete** symbol in the command bar.

Illustration 60: Delete the byte at its place of origin.

➡ Choose the desired module under **Left side bar | Library**.

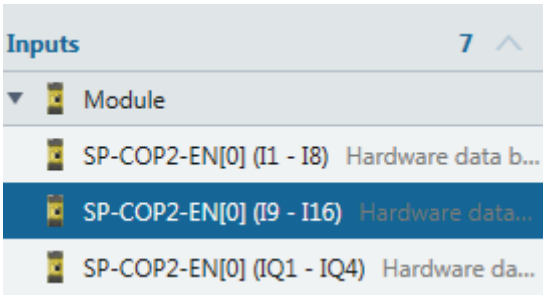


Illustration 61: Module in the library

➡ Use the mouse button to drag the module into the work area on **byte 23**.

Illustration 62: Module on new byte

11.5.2 Data set 2

Data set 2 (32 bytes) contains the check values (CRCs) for the system configuration.

Configuration check values (CRCs)

Data set 2 contains the following configuration check values for the samosPRO COMPACT system:

- Project-CRC of the project file created with samosPLAN5+.
- System-CRC, uniquely assigned to a module version, consisting of internal software and hardware version

The CRCs are each 4 Bytes in length. Data set 2 can be read only.

The project CRC is transmitted in Big Endian format.

Table 111: Output data set 2 samosPRO COMPACT to --> SP-EN-ETC

Byte	assignment
Byte 0	Project CRC Value is on the first page in the project report from samosPLAN5+. Example: CRC Station 1: 0x2ac78506
Byte 1	
Byte 2	
Byte 3	
Byte 4	System CRC Caution: Review of this system CRC requires a custom material number and thus a special agreement with Wieland Electric.
Byte 5	
Byte 6	
Byte 7	
Byte 8 to byte 31	Reserved for as yet unknown applications
Length	**32 bytes**

11.5.3 Data set 3

Data set 3 (60 bytes) contains the state and diagnostic data for the various modules, with four (4) bytes per module, with the main module comprising 3 x 4 bytes. For details see table *Meaning of module state bits of EA modules SP-SDI and SP-SDIO [ch. 3.3.5, p. 20]*.

Error and state information for the modules

Data set 3 contains the state information for the modules that will be transferred to the network.

Ten bytes are transferred for each SP-COPx main module. For each SP-SDI and SP-SDIO I/O module, four bytes are transmitted in the little endian format, e.g. as a 32-bit word, with the first byte being placed into the least significant byte of the whole number (extreme left) and the fourth byte into the most significant byte of the whole number (extreme right).

Data set 3 cannot be changed.

NOTICE

- Reserved (for future use) = static 1 (no state change)
- Not used (can be 0 or 1), both values occur.
- If there is no module, all values - including the reserved values - are set to logical 1.

Table 112: Output data set 3 samosPRO COMPACT to --> SP-EN-ETC

Byte	assignment
Byte 0	Module state SP-COPx
Byte 1	Module state SP-COPx
Byte 2	Test impulse comparison for SP-COP inputs
Byte 3	Test impulse comparison for SP-COP inputs
Byte 4	Test impulse comparison for SP-COP inputs
Byte 5	State of two-channel SP-COP inputs
Byte 6	State of two-channel SP-COP inputs
Byte 7	Reserved
Byte 8	Stuck-at error at SP-COP outputs
Byte 9	Stuck-at error at SP-COP outputs
Byte 10	Reserved
Byte 11	Reserved
Byte 12	Status of Module 1
Byte 13	Status of Module 1
Byte 14	Status of Module 1
Byte 15	Status of Module 1
Byte 16	Status of Module 2
Byte 17	Status of Module 2
Byte 18	Status of Module 2
Byte 19	Status of Module 2
Byte 20	Status of Module 3
Byte 21	Status of Module 3
Byte 22	Status of Module 3
Byte 23	Status of Module 3
Byte 24	Status of Module 4
Byte 25	Status of Module 4
Byte 26	Status of Module 4

Byte	assignment
Byte 27	Status of Module 4
Byte 28	Status of Module 5
Byte 29	Status of Module 5
Byte 30	Status of Module 5
Byte 31	Status of Module 5
Byte 32	Status of Module 6
Byte 33	Status of Module 6
Byte 34	Status of Module 6
Byte 35	Status of Module 6
Byte 36	Status of Module 7
Byte 37	Status of Module 7
Byte 38	Status of Module 7
Byte 39	Status of Module 7
Byte 40	Status of Module 8
Byte 41	Status of Module 8
Byte 42	Status of Module 8
Byte 43	Status of Module 8
Byte 44	Status of Module 9
Byte 45	Status of Module 9
Byte 46	Status of Module 9
Byte 47	Status of Module 9
Byte 48	Status of Module 10
Byte 49	Status of Module 10
Byte 50	Status of Module 10
Byte 51	Status of Module 10
Byte 52	Status of Module 11
Byte 53	Status of Module 11
Byte 54	Status of Module 11
Byte 55	Status of Module 11
Byte 56	Status of Module 12
Byte 57	Status of Module 12
Byte 58	Status of Module 12
Byte 59	Status of Module 12
Length	**60 bytes**

11.6 Data received from the network

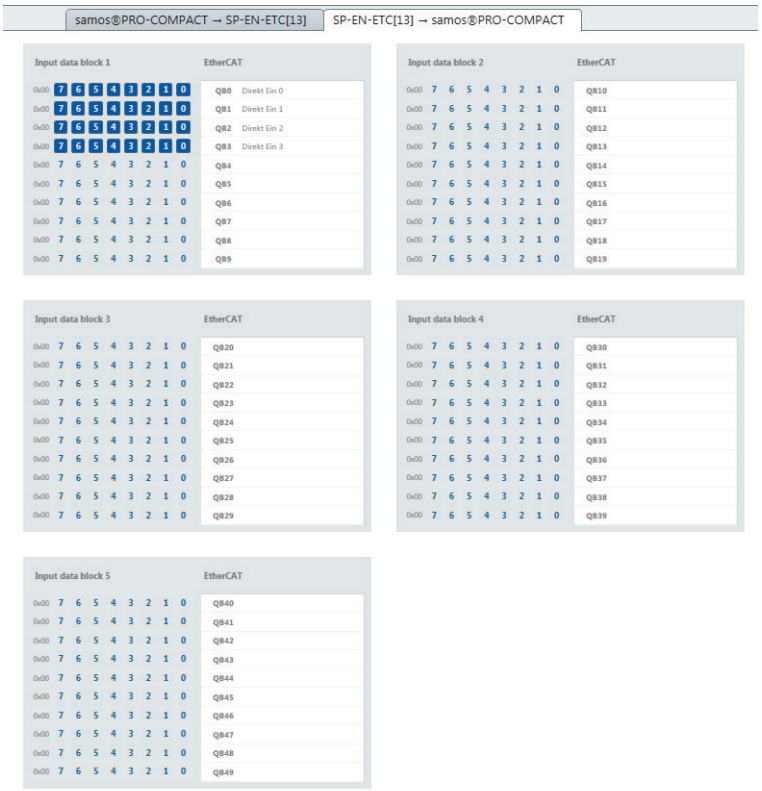
The data received from the network is divided into five data blocks of 10 bytes each for clarity.

The content of the input data blocks can be used in the logic editor of the samosPRO COMPACT CPU, as well as made available for another network via a second gateway in the samosPRO COMPACT gateway within the samosPRO COMPACT system.

NOTICE

- In order to use network data in the logic editor or as input for another network, you must assign a tag name for each bit to be used.
- Bits without specific tag names will not be available in the logic editor or for routing via a second gateway. Detailed information about how to assign tag names for the data received may be found in the corresponding sections of the chapters on the various gateways.
- You can monitor current communication with the network with the aid of input data state bits for receiving data from the network and the output data state bit for transmitting data to the network in the logic editor. When the gateway detects a communication error, both the content of the data sets and the associated state bit are set to zero (logical 0).
- When all communication fails, the data of the output data sets and the input data state bit are set to zero (logical 0).
- When a connection is closed while others remain available, the LED MS or LED state will flash red/green for a total of 10 seconds and an entry will be made in the error log. In this case the state bits are not affected.

Tag names pre-assigned in samosPLAN5+ for the EtherCAT gateway



Delete any bytes not required

You can delete bytes pre-allocated by samosPLAN5+ that you do not require by clicking on them with the mouse.

➡ Start samosPLAN5+.
➡ Import the hardware configuration, including for the SP-EN-ETC gateway.
Instructions: "samosPLAN5+ Software" manual, section "Mit dem samosPRO-System verbinden"
➡ Switch to the **Gateway** view.

➡ Click on the byte you do not need and wish to delete.

➡ Click on the **Delete** icon in the command bar.

You will find further information about how to configure the process diagram here:

- *Configuration of gateways with samosPLAN5+ [ch. 5, p. 31]*
- "samosPLAN5+ Software" manual (BA000967)

Structure of the data block

The input data block consists of 50 bytes (byte 0 to 49) of data that is transferred from the EtherCAT network to the SP-EN-ETC gateway. The content of the data bytes does not meet the requirements of a safety system. The values are only current as long as the gateway to the EtherCAT network is connected and the gateway status is **Operational**. As soon as the state machine of the gateway adopts a state other than **Operational**, these data are set to zero.

Also see: *Gateway state machine [ch. 11.3, p. 155]*

Table 113: Input data block 1–5 of the SP-EN-ETC to --> samosPRO -COMPACT

	Input data block 1	Input data block 2	Input data block 3	Input data block 4	Input data block 5
Byte 0	Byte 0	Byte 10	Byte 20	Byte 30	Byte 40
Byte 1					
Byte 2					
Byte 3					
Byte 4					
Byte 5					
Byte 6					
Byte 7					
Byte 8					
Byte 9	Byte 9	Byte 19	Byte 29	Byte 39	Byte 49
Length	10 bytes	10 bytes	10 bytes	10 bytes	10 bytes

11.7 Configuring an EtherCAT network

A device description file (ESI = **E**therCAT **S**lave **I**nformation) is delivered with the SP-EN-ETC. The EtherCAT master integrates this file into the EtherCAT system so that the master has the necessary EtherCAT configuration data and can establish a connection to the gateway.

Please read the manual depending on your controller to see which steps are required in detail.

11.8 EtherCAT configuration of the gateway - how the data are transferred

The following steps are required to configure the communication between the PLC programming system and the gateway. Configuration in the programming system is done by integrating a standardized ESI description file.

NOTICE

This documentation does not address the installation of the EtherCAT network or the other components of the automation system project in the network configuration tool. It is assumed that the EtherCAT project in the configuration program (e.g. Beckhoff TwinCAT) has already been set up. The examples presented are based on configurations created with the help of Beckhoff TwinCAT.

Step 1: Install the EtherCAT slave description file

Before the SP-EN-ETC can be used for the first time as part of the network configuration tool (e.g. Beckhoff TwinCAT), the gateway description file must first be installed in the hardware catalog of the tool.

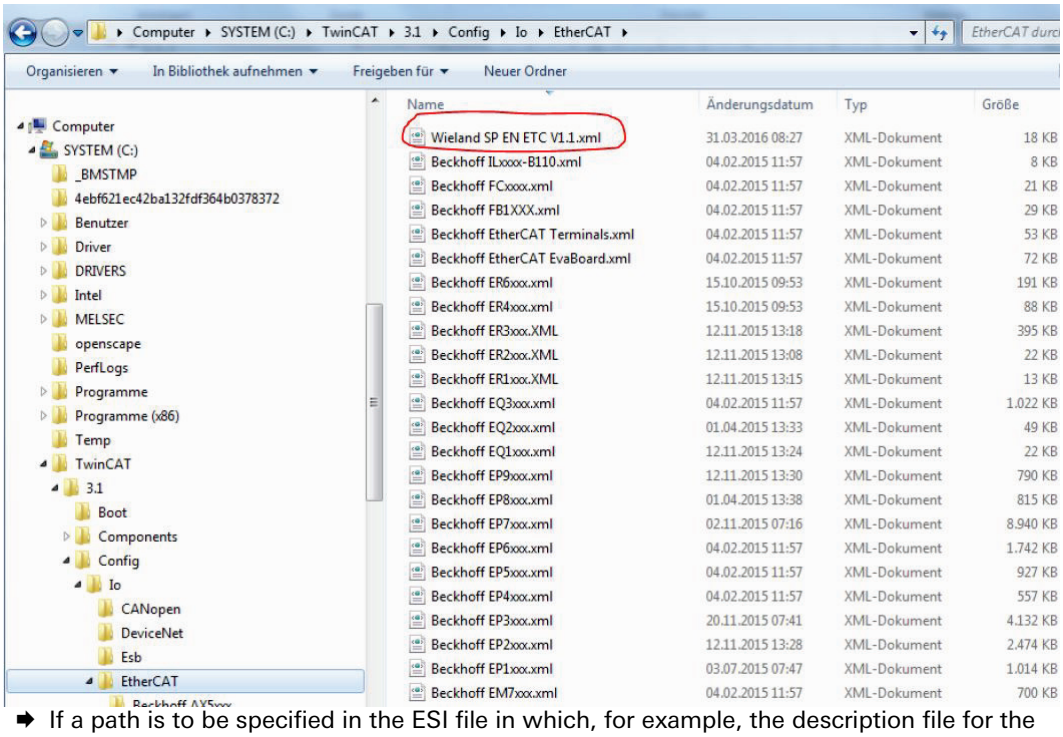
- ➡ Download the GSD file and the equipment symbol from the product page for the SP-EN-ETC (http://eshop.wieland-electric.com).
- ➡ Follow the instructions to install XML in the online help or user manual of the EtherCAT network configuration tool for the master or for the EtherCAT control system.

Step 2: Add the gateway to a PLC project

To make the system data for the samosPRO COMPACT system available in the process diagram of the PLC, the gateway must first be added to the hardware configuration. The procedure to be used depends on the hardware configuration software of the PLC used. Please also read the documentation for the corresponding software in this regard.

The example below shows how the gateway is added to a control project in Beckhoff TwinCAT.

➡ Copy the description file **Wieland SP EN ETC V1.1.xml** into the TwinCAT folder. An example of a typical installation can be seen below:

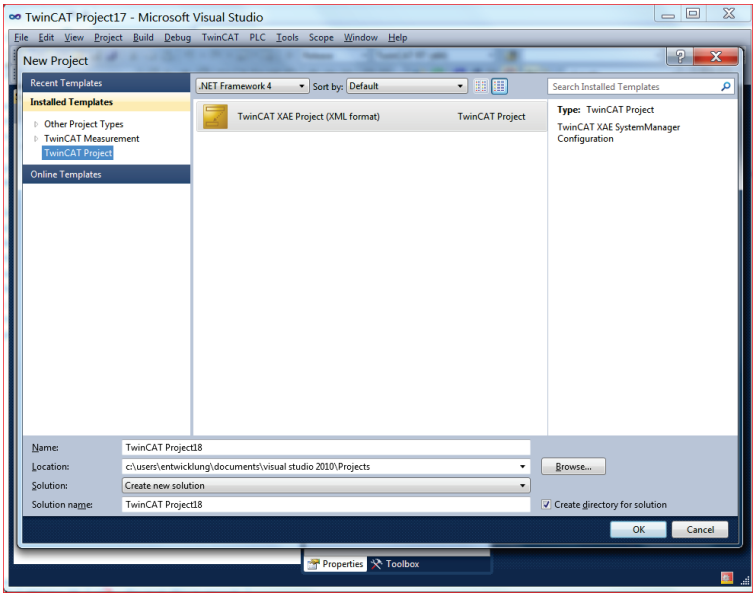


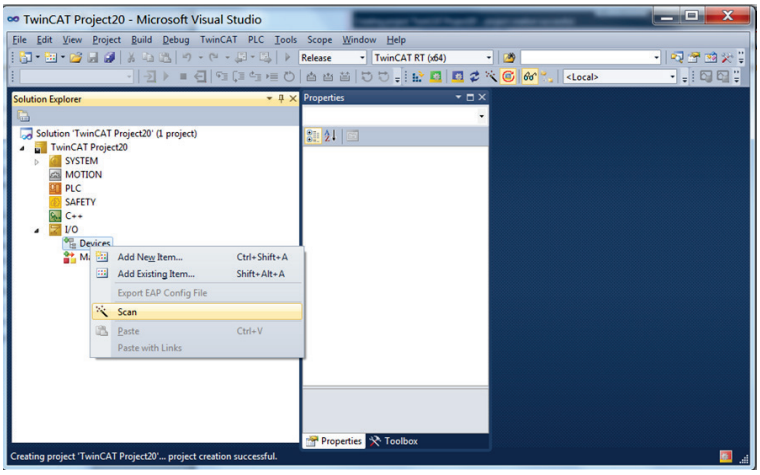
➡ If a path is to be specified in the ESI file in which, for example, the description file for the expansion modules is locates, create this path in the directory exactly as described in the file.

➡ Re-start TwinCAT.
Note: The folder with the current description files is only read when the program is restarted.

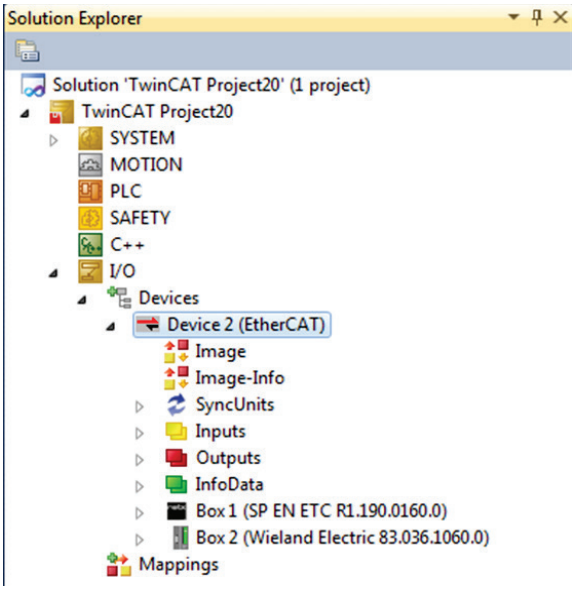
Example: This is not true for the gateway, but is important for other slaves (e.g. ricos FLEX EtherCAT slave).

Step 3: Create a new project





After you have connected the controller – i.e. the EtherCAT master to the EtherCAT slaves – you can scan the connected slaves.



TwinCAT shows the found slaves in the Solution Explorer as a box with the corresponding device names.

Table 114: Error

Error	Cause
Box is displayed with no device names.	ESI file was not found.
No EtherCAT slave (box) is displayed.	Modules are not connected to the EtherCAT master or are not powered.
The input data (inputs) are not up to date or all have the value 0.	SP-COP2 is at stop or no data were mapped to output data set 1. Data set 2 and 3 are displayed once SP-COP2 is in the RUN state.
The output data (outputs) are transmitted to the gateway but not displayed in the input data sets.	No tags have been created in the input data set.

11.9 Diagnostic LEDs on the gateway and troubleshooting

Information on the diagnostics of the samosPRO COMPACT system can be found in the "samosPLAN5+ Software" manual (BA000967).

Table 115: Troubleshooting for the SP-EN-ETC

Error		Possible cause	Possible remedy
Key: ○ LED off / ✳ LED flashes / ● LED lights up			
The samosPLAN5+ cannot establish a link to the samosPRO COMPACT gateway.			• Switch on the power supply. • Check the communication settings in samosPLAN5+.
The SP-EN-ETC is not providing any input data.		• EtherCAT not connected.	• Connect RJ45 cable to ECAT-IN.
MS LED	✳ / ✳ Red / green		
The SP-EN-ETC is not providing any input data.		• RJ45 is connected to the port, no data on the EtherCAT Net.	• Activate EtherCAT.
MS LED	✳ / ✳ Red / green		
LINK	● Green		
The SP-EN-ETC is not providing any input data.		• RJ45 is connected to the port, EtherCAT not active.	• Activate EtherCAT and initialize Gateway. **Init** state
MS LED	✳ / ✳ Red / green		
LINK	✳ Green		
Controller errors		• Incorrect EtherCAT configuration, gateway is addressed with incorrect data. • Gateway is in **Pre-Op** state.	• Check network and device configuration. • Switch power off and back on.
MS LED	✳ Red		
RUN	✳ Green / flashing		
ERR	✳ Red / flashing		
LINK	✳ Green		

Error		Possible cause	Possible remedy
The SP-EN-ETC is not providing any input data.		• Gateway state is **Init**.	• Switch EtherCAT to **Op** state.
MS LED	 Red / green		
LINK	 Green		
LINK	 Green		
The SP-EN-ETC is not providing any input data.		• Gateway state is **Pre-Op**.	• Switch EtherCAT to **Op** state.
MS LED	 Red / green		
RUN	 Green		
LINK	 Green		
The SP-EN-ETC is not providing any input data.		• Gateway state is **Safe-Op**.	• Switch EtherCAT to **Op** state.
MS LED	 Red / green		
RUN	 Green/flash		
LINK	 Green		
The SP-EN-ETC is not providing any input data.		• No EtherCAT data, but there is a bus connection to next EtherCAT slave.	• Re-start EtherCAT master or supply master with power. • Check RJ45 cable. • Repair interruption to the EtherCAT network.
MS LED	 Red / green		
RUN	 Green/flash		
ERR	 Red / double flash		
RUN	 Green/flash		
ERR	 Red / double flash		

Notes on troubleshooting

- LINK LEDs

  Use the state of the LINK LEDs to check whether there is a connection to the Ethernet.

- Cables

  Check that the pin assignment of the used cable is correct.

- Configuration

  Make sure that the gateway is installed right next to the samosPRO COP and that no more than two samosPRO gateways are connected. Also note that only a maximum of 12 expansion modules are connected to the samosPRO COP.

- Mechanical strength

  Check whether the RJ 45 connectors are engaged by gently pulling on the EtherCAT connection cables.
  In case of high mechanical load, secure the RJ45 cable with a tension relief.

12 Technical data

12.1 Modbus TCP, PROFINET IO and EtherNet/IP gateway

Use the SP-COP2-ENI module for the Modbus TCP, PROFINET IO and EtherNet/IP functionalities.

You will find the technical data for this module here:
"samosPRO Hardware" manual, (BA000966)", section "Module SP-COP1 and SP-COP2-ENx"

12.2 EtherCAT gateway

Interface	Minimal	Typical	Maximum
Field bus	EtherCAT		
Connection technology	RJ45 socket		
Transfer rate	100 Mbit/s (100 Base-TX)		
Device type	EtherCAT slave		
Data length: Inputs	50 bytes from EtherCAT to samosPRO COMPACT		
Data length: Outputs	142 bytes (50 + 32 + 60) from samosPRO COMPACT to EtherCAT		
galvanic isolation	Yes - between EtherCAT (RJ45) and system voltage		
Type of insulation	Function insulation		
Field bus	EtherCAT		

12.3 PROFIBUS DP

Interface	Minimal	Typical	Maximum
Field bus	PROFIBUS-DP-V0		
Interface level	RS-485		
Connection technology	9-pin D-sub socket		
Slave address (set via rotary switch)	0		99
Slave address (set in samosPLAN5+[1)]	3		125
Baud rate (automatic adaptation)			12 MBaud
Baud rate (kbits/s with standard line)			Maximum line length
9.6/19.2/93.75			1200 m
187.5			1000 m
500			400 m
1.500			200 m
12.000			100 m
Line parameters	see *PROFIBUS-DP gateway [ch. 9, p. 107]*		
[1)] To set the slave address via software, the hardware address setting must be "0".			

12.4 CANopen gateways

Interface	Minimal	Typical	Maximum
Field bus	CANopen DS-301		
Interface level	RS-485		
Connection technology	5-pin "open style" socket		
Slave address (set via rotary switch)	0		99
Slave address (set in samosPLAN5+ [1)]	1		127
Baud rate (kbits/s with standard line)			Maximum line length
125			500 m
250			250 m
500			100 m
800			40 m
1000			20 m
Line parameters	see *CANopen gateway [ch. 10, p. 118]*		
[1)] To set the slave address via software, the hardware address setting must be "0".			

12.5 Technical data for supply circuit

These technical data apply to all gateway modules.

Supply circuit (e.g. via SBUS+)	Minimal	Typical	Maximum
Supply voltage	16.8 V DC	24 V DC	30 V DC
Power consumption			2.4 W

12.6 General technical data

These technical data apply to all gateway modules.

General technical data	
Connecting terminals	
Field bus	See: *Interfaces and operation [ch. 9.1, p. 107]*
SBUS+	10-pin plug for internal safety bus
Climatic conditions	
Environmental operating temperature T_A	-25 to +55°C
Storage temperature	-25 to +70°C
Relative humidity	10 to 95%, non-condensing
Climatic conditions (EN 61131-2) Air pressure during operation	 860 to 1060 hPa
Mechanical strength	
Sinus vibrations (EN 60068-2-6) Frequency range Amplitude Acceleration Number of cycles	 5 to 150 Hz 3.5 mm (5 to < 9 Hz) 1 g (9 to 150 Hz) 10 per axis (on 3 axes)
Broad-band noise vibration (EN 60068-2-64) Frequency range Acceleration	 10 to 500 Hz 5 g
Half-sinusoidal impulses (EN 60068-2-27) Acceleration Duration	 15 g 11 ms
Electric safety	See SP-COPx
Protective type (EN 60529)	IP 20
Protection class	III
Electromagnetic compatibility	EN 61000-6-2/EN 55011 Class A
Mechanics and set-up	
Housing material	Polycarbonate
Housing type	Device for installation in switch box
Housing protection type/terminals	IP 20/IP 40
Color Gateways	 Light gray
Weight	0.16 kg
SBUS+ connection (internal bus) Number of pins Gateways	 10 1 plug on the left and 1 plug on the right
Hat rail	Hat rail according to EN 60715

12.7 Dimensional drawings

12.7.1 SP-COP1-xxx / SP-COP2-xxx controller modules

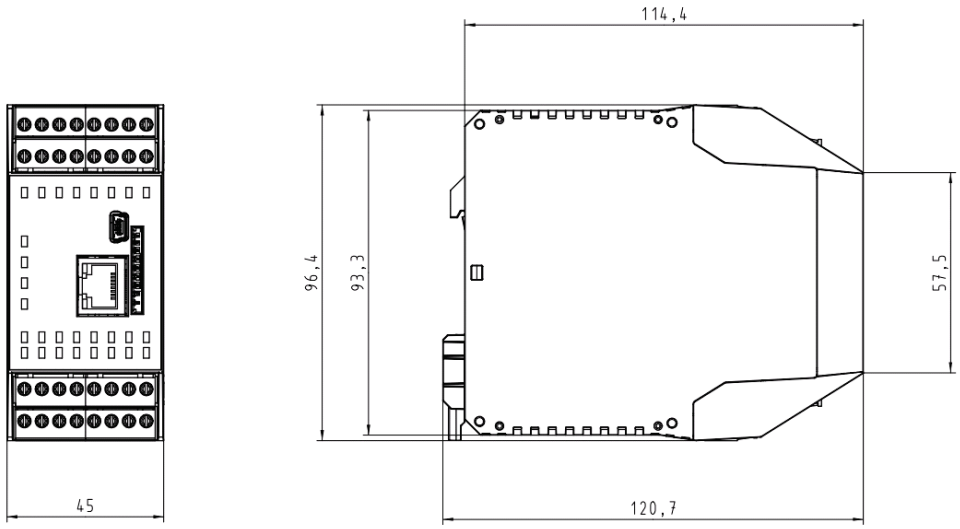


Illustration 63: SP-COP1 / SP-COP2 dimensions (mm)

12.7.2 SP-CANopen and SP-PROFIBUS-DP

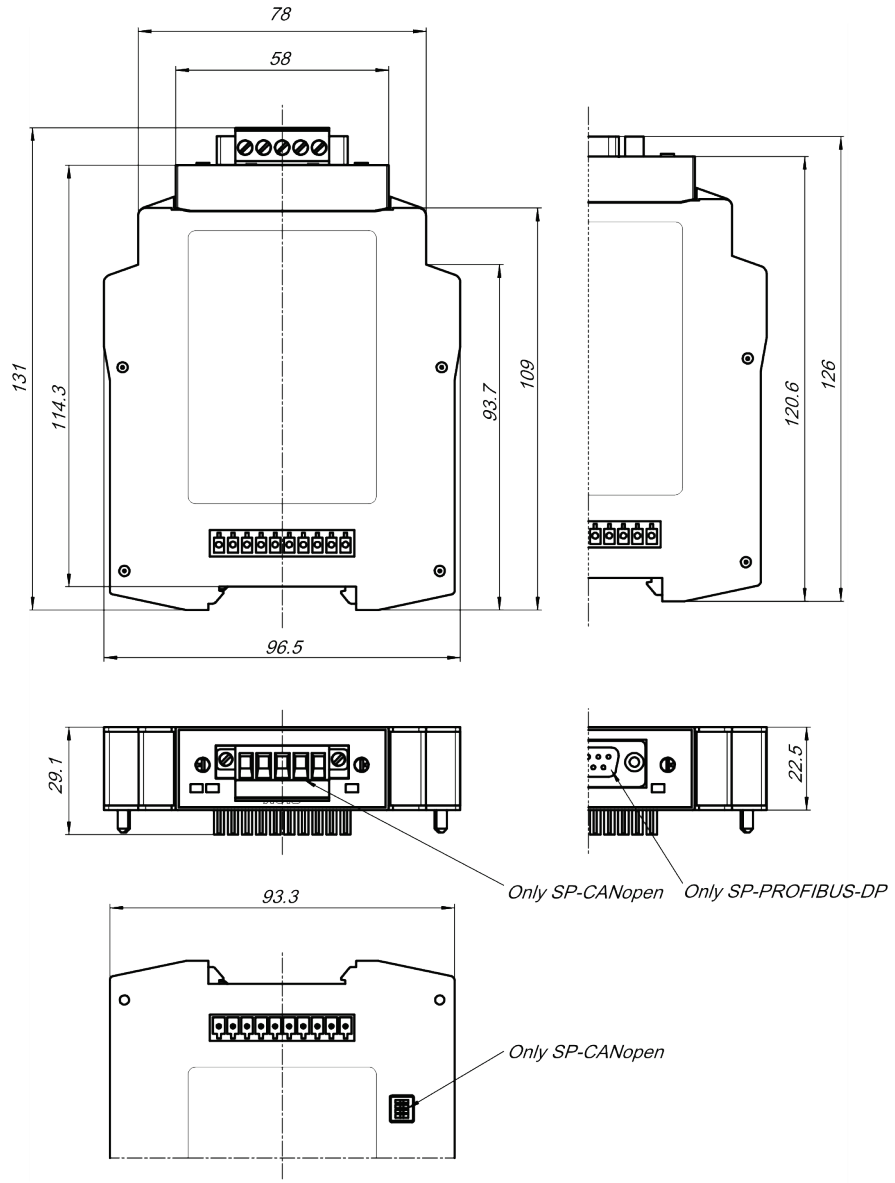


Illustration 64: Dimensional drawing SP-CANopen and SP-PROFIBUS-DP (mm)

12.7.3 EtherCAT gateway (SP-EN-ETC)

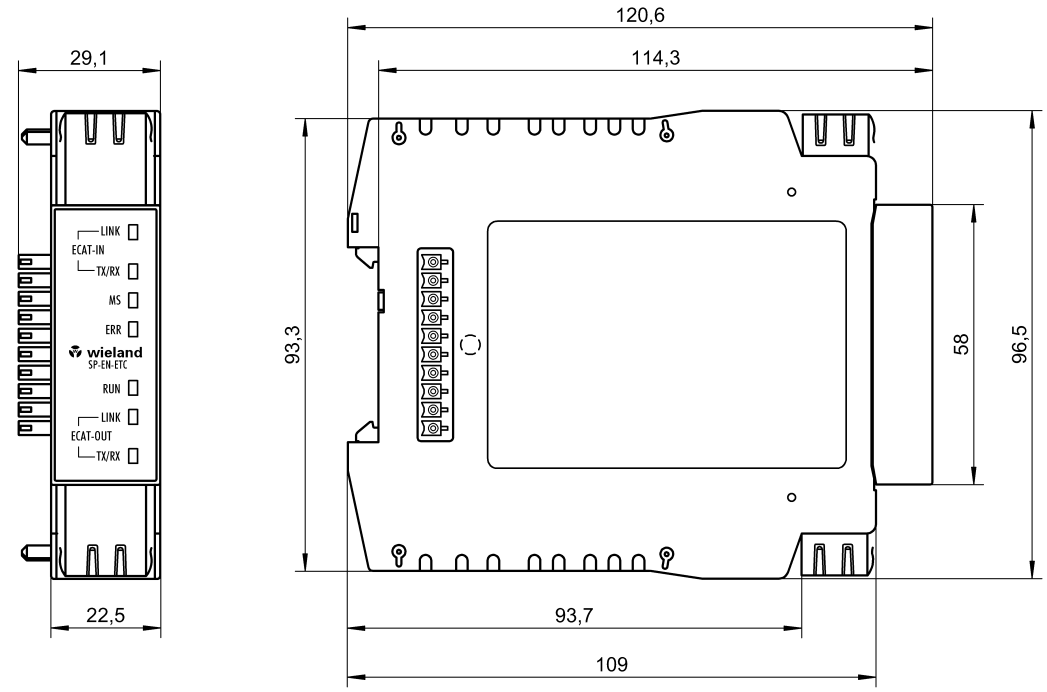


Illustration 65: Dimension drawing of EtherCAT gateway SP-EN-ETC (mm)

13 Order data

13.1 samosPRO – COMPACT – modules and accessories

Table 116: Part numbers for samosPRO safety control modules with COMPACT modules

Type	Description	Part number
SP-COP1-A	COMPACT module, USB connection, 20 inputs / 4 outputs Screw terminals, pluggable	R1.190.1110.0
SP-COP1-C	COMPACT module, USB connection, 20 inputs / 4 outputs Spring-loaded terminals, pluggable	R1.190.1120.0
SP-COP1-P-A	COMPACT module, USB connection, 20 inputs / 4 outputs with press function Screw terminals, pluggable	R1.190.1130.0
SP-COP1-P-C	COMPACT module, USB connection, 20 inputs / 4 outputs with press functions Spring-loaded terminals, pluggable	R1.190.1140.0
SP-COP2-EN-A	COMPACT module, USB and Ethernet connection, 16 inputs / 4 outputs and 4 configurable inputs or outputs Screw terminals, pluggable	R1.190.1210.0
SP-COP2-EN-C	COMPACT module, USB and Ethernet connection, 16 inputs / 4 outputs and 4 configurable inputs or outputs Spring-loaded terminals, pluggable	R1.190.1220.0
SP-COP2-EN-P-A	COMPACT module, USB and Ethernet connection, 16 inputs / 4 outputs and 4 configurable inputs or outputs with press functions Screw terminals, pluggable	R1.190.1230.0
SP-COP2-EN-P-C	COMPACT module, USB and Ethernet connection, 16 inputs / 4 outputs and 4 configurable inputs or outputs with press functions Spring-loaded terminals, pluggable	R1.190.1240.0
SP-COP2-ENI-A	COMPACT module, USB and industrial Ethernet connection, 16 inputs / 4 outputs and 4 configurable inputs or outputs Screw terminals, pluggable	R1.190.1310.0

Type	Description	Part number
SP-COP2-ENI-C	COMPACT module, USB and industrial Ethernet connection, 16 inputs / 4 outputs and 4 configurable inputs or outputs Spring-loaded terminals, pluggable	R1.190.1320.0
SP-COP2-ENI-P-A	COMPACT module, USB and industrial Ethernet connection, 16 inputs / 4 outputs and 4 configurable inputs or outputs with press functions Screw terminals, pluggable	R1.190.1330.0
SP-COP2-ENI-P-C	COMPACT module, USB and industrial Ethernet connection, 16 inputs / 4 outputs and 4 configurable inputs or outputs with press functions Spring-loaded terminals, pluggable	R1.190.1340.0
SP-COP-CARD1	Program removable storage	R1.190.1000.0
SP-CABLE-USB1	1.8 m USB configuration capable	R1.190.1010.0
SP-CABLE-ETH1	2 m Ethernet configuration capable	R1.190.1020.0
SP-PLAN5+	CD with samosPLAN5+ programming software and manuals	R1.190.1030.0
SP-COP-STARTER-SET	The set contains one each of SP-COP2-EN-A, SP-SDIO, SP-COP-CARD1, SP-PLAN5+, SP-CABLE-USB1	R1.190.1100.0
SP-CANopen	CANopen gateway	R1.190.0210.0
SP-PROFIBUS-DP	PROFIBUS-DP gateway	R1.190.0190.0
SP-SDIO84-P1-K-A	Input/output expansion with output test pulses 8 inputs/4 outputs Screw terminals, pluggable	R1.190.0030.0
SP-SDIO84-P1-K-C	Input/output expansion with output test pulses 8 inputs/4 outputs Spring-loaded terminals, pluggable	R1.190.0040.0
SP-SDI8-P1-K-A	Input expansion 8 inputs Screw terminals, pluggable	R1.190.0050.0
SP-SDI8-P1-K-C	Input expansion 8 inputs Spring-loaded terminals, pluggable	R1.190.0060.0
SP-DIO84-P1-K-A	Input/output expansion 4 inputs / 4 outputs and 4 configurable inputs or outputs Screw terminals, pluggable	R1.190.1050.0
SP-DIO84-P1-K-C	Input/output expansion 4 inputs / 4 outputs and 4 configurable inputs or outputs Spring-loaded terminals, pluggable	R1.190.1060.0
SP-EN-ETC	EtherCAT Gateway	R1.190.0160.0
WKFN 2.5 E/35 GO-URL	fasis series level terminal with diode	56.703.8755.9
APFN 2.5 E/35	Terminal plate for WKFN 2.5 E/35	07.312.7355.0

13.2 Modules for contact expansion

Type	Description	Part number
SNE 1	Forcibly actuated single relay, 24 V DC, plug socket, 2 changeovers	R1.188.3950.0
SNE 4004K-A	Contact expansion, 24 V DC, 4 NC (normally closed contact), 3 NO (normally open contact), Screw terminals, pluggable	R1.188.0590.0
SNE 4004K-C	Contact expansion, 24 V DC, 4 NC (normally closed contact), 3 NO (normally open contact), Spring-loaded terminals, pluggable	R1.188.1980.0
SNE 4012K-A	Contact expansion, 24 V DC, 2 NC (normally closed contact), 1 NO (normally open contact), Screw terminals, pluggable	R1.188.3910.0
SNE 4012K-C	Contact expansion, 24 V DC, 2 NC (normally closed contact), 1 NO (normally open contact), Spring-loaded terminals, pluggable	R1.188.3920.0
SNE 4024K-A	Contact expansion with 2 relay groups, 24 V DC, 2 x 2 NC (normally closed contact), 2 x 1 NO (normally open contact), Screw terminals, pluggable	R1.188.3930.0
SNE 4024K-C	Contact expansion with 2 relay groups, 24 V DC, 2 x 2 NC (normally closed contact), 2 x 1 NO (normally open contact), Spring-loaded terminals, pluggable	R1.188.3940.0
SNE 4028S-A	Contact expansion, 24 V DC, 8 NC (normally closed contact), 2 NO (normally open contact), Screw terminals, pluggable	R1.188.3120.0
SNE 4028S-C	Contact expansion, 24 V DC, 8 NC (normally closed contact), 2 NO (normally open contact), Screw terminals, pluggable	R1.188.3540.0