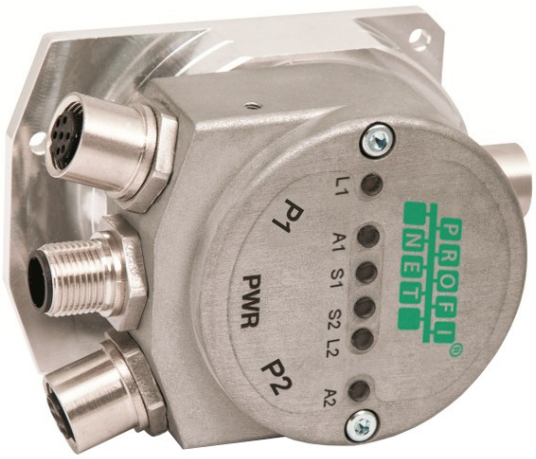
Smart encoders & actuators

User's guide

IF55 ROT PT
IF55 LIN PT



- SSI to Profinet converter
- Suitable for SSI rotary and linear encoders
- Accept MSB & LSB Aligned protocols up to 30 bits
- M12 connectors
- Encoder Profile Specifications V4.1 version 3.162
- RT real-time transmission & IRT isochronous real-time mode

Suitable for the following models:

- IF55 ROT PT
- IF55 LIN PT

General Contents

Table of contents

Subject Index

Table of figures

Typographic and iconographic conventions

In this guide, to make it easier to understand and read the text the following typographic and iconographic conventions are used:

- parameters and objects of both Lika device and interface are coloured in **GREEN**;
- alarms are coloured in **RED**;
- states are coloured in **FUCSIA**.

When scrolling through the text some icons can be found on the side of the page: they are expressly designed to highlight the parts of the text which are of great interest and significance for the user. Sometimes they are used to warn against dangers or potential sources of danger arising from the use of the device. You are advised to follow strictly the instructions given in this guide in order to guarantee the safety of the user and ensure the performance of the device. In this guide the following symbols are used:

	This icon, followed by the word **WARNING**, is meant to highlight the parts of the text where information of great significance for the user can be found: user must pay the greatest attention to them! Instructions must be followed strictly in order to guarantee the safety of the user and a correct use of the device. Failure to heed a warning or comply with instructions could lead to personal injury and/or damage to the unit or other equipment.
	This icon, followed by the word **NOTE**, is meant to highlight the parts of the text where important notes needful for a correct and reliable use of the device can be found. User must pay attention to them! Failure to comply with instructions could cause the equipment to be set wrongly: hence a faulty and improper working of the device could be the consequence.
	This icon is meant to highlight the parts of the text where suggestions useful for making it easier to set the device and optimize performance and reliability can be found. Sometimes this symbol is followed by the word **EXAMPLE** when instructions for setting parameters are accompanied by examples to clarify the explanation.

Preliminary information

This guide is designed to provide the most complete and exhaustive information the operator needs to correctly and safely install and operate the following **SSI to Profinet gateways of the IF55 series**:

IF55 LIN PT **(DAP 1 : suitable for linear encoders with resolution 1,000,000 nm -1 mm- down to 1 nm, see the Physical pulse resolution [nm] parameter; max. number of bits: 30, see the Physical Total resolution [bit] parameter)**

IF55 ROT PT **(DAP 2 : suitable for encoders with singleturn resolution up to 30 bits, see the Physical singleturn resolution [bit] parameter; and multiturn resolution -number of revolutions- up to 29 bits, see the Physical multiturn resolution [bit] parameter; max. number of bits: 30, Physical singleturn resolution [bit] + Physical multiturn resolution [bit] ≤ 30)**

IF55 series gateways allow the **integration of SSI encoders**, both rotary and linear, **into conventional fieldbuses or industrial Ethernet networks**.
The present manual is specifically designed to describe the SSI to Profinet IF55 model for rotary and linear encoders (order codes: IF55 ROT PT and IF55 LIN PT).
For information on the gateways designed for the integration with other protocols (for example: SSI to Profibus: order codes IF55 ROT PB and IF55 LIN PB; SSI to CANopen: order codes IF55 ROT CB and IF55 LIN CB; etc.), refer to the specific documentation.

Please note that the present manual does not prescind from the user's guide of the SSI encoder the gateway has to be connected to. Please read carefully the encoder's documentation before installing, connecting and operating the measuring system.
For detailed technical specifications please refer also to the product datasheet.

To make it easier to read the text, this guide can be divided into some main sections.
In the first section (from chapter 1 to chapter 4) general information concerning the safety, the mechanical installation and the electrical connection.
In the second section (chapter 5) information on how to install and configure the encoder in the TIA development environment as well as tips for setting up and running properly and efficiently the unit are provided.
In the third section (from chapter 6 to chapter 12) both general and specific information is given on the Profinet interface. In this section the interface features and the parameters implemented in the unit are fully described.
In the last section (from chapter 13 to chapter 15) some examples of programming and advanced maintenance information are explained.

Glossary of Profinet terms

PROFINET IO, like many other networking systems, has a set of unique terminology. Table below contains a few of the technical terms used in this guide to describe the PROFINET IO interface. Sometimes they also refer more specifically to the S7 programming environment. They are listed in alphabetical order.

Acyclic Communications	Unscheduled, on demand communications. Diagnostic messages from an IO Supervisor to an IO Device are Acyclic. Refer to page 79.
AP	Application Process - The application process running in the device. PROFINET supports a default Application Processes and additional profile specific application processes.
API	The value of the API (Application Process Identifier) parameter specifies the application that is processing the IO data. PROFINET standard IEC 61158 assigns profiles to certain APIs (PROFIdrive, PROFIslave) which are defined by the PROFINET User Organization. The standard API is 0.
Application class	An application class specifies a number of mandatory functions and addition optional functions to be supported by an IO device. The Profinet encoders can be configured as CLASS 3 and CLASS 4 PROFINET IO devices according to the encoder profile. Refer to page 66.
AR	Application Relation - The relationship between a PROFINET IO Controller and an IO device. A PROFINET IO device can support more than one Application Relationship.
Bus	A bus is a communication medium connecting several nodes. Data can be transferred via serial or parallel circuits, that is, via electrical conductors or fiber optic.
Channel	A single IO point. A Channel can be discrete or analog.
Consumer Status	The Status an IO device provides to an IO Controller for the data it consumes from IO Controller.
CR	Communication Relationship - A virtual communication channel within an AR.
Cyclic Communications	Scheduled, repetitive communications. IO data and alarm transfers are cyclic.
DAP (Device Access Point)	Module of the GSDML file which is intended to describe the parameters that are specific to a device.
Data block	In contrast to code blocks, data blocks (DB) do not contain Step 7 statements. They are used to save data, i.e. variable data which are processed by the user program. Global data blocks serve to accommodate user data which can be used by all other blocks.

DCP	Discovery Control Protocol - A communications protocol with PROFINET IO that allows an IO Controller or Supervisor to find every PROFINET IO device on a subnet.
Determinism	Determinism means that a system responds in a predictable (deterministic) manner.
Device Access Point (DAP)	Module of the GSDML file which is intended to describe the parameters that are specific to a device.
Device name	Before an IO device can be addressed by an IO controller, it must have a device name. In PROFINET, this method was selected because it is simpler to work with names than with complex IP addresses.
Encoder Profile	The PROFINET profile for Encoders is intended to define a standard application interface for encoders. The profile is a supplement to the PROFIdrive profile, so it is mandatory to read the PROFIdrive profile before implementing the encoder profile. Profinet encoders from Lika Electronic comply with the Encoder Profile Specifications V4.1 version 3.162. See also "Profile".
Function	Functions (FC) are code blocks which can be programmed by the user. A FC does not have a "memory". Temporary variables as well as parameters transferred to the function when the latter is called are saved in a L stack. They are lost following processing of the FC.
Function block	Function blocks (FB) are code blocks with a "memory" which are programmed by the user. They have an assigned instance data block (instance DB) as memory. Parameters transferred to a FB as well as the static variables are saved in this data block. An FB contains a program which is always executed when the FB is called by another code block. Function blocks facilitate the programming of frequently repeated, complex functions.
Frame ID	The two-byte field in the Ethernet frame which defines the type of PROFINET IO message.
GSD	The properties of a PROFINET device are described in a GSD file (General Station Description) that contains all the information required for configuration. In PROFINET IO, the GSD file is in XML format. The structure of the GSD file conforms to ISO 15734, which is the world-wide standard for device descriptions. Refer to page 50.
GSDML	General Station Description Markup Language – The file containing the XML description of the PROFINET IO device. Refer to page 50.
IO Controller	Device used to address the connected IO devices. This means that the IO controller exchanges input and output signals with assigned field devices. The IO controller is often the controller on which the automation program runs. Refer to page 64.
IO Device	A decentralized field device that is assigned to one of the IO controllers (e.g. remote IO, encoders, valve terminals,

	frequency converters, switches, etc.). Refer to page 64.
IO Parameter Server	An IO Parameter Server is a server station, usually a PC, for loading and saving the configuration data (records) of IO Devices.
IO Supervisor	Programming device, PC or HMI device used for commissioning and diagnostics of IO Controllers and IO Devices. Refer to page 64.
IP address	The IP address is the name of the unit in a network using the Internet protocol. Refer to page 33.
IRT	Synchronized transmission procedure for the cyclic exchange of IRT data between PROFINET devices. A reserved bandwidth within the send clock is available for the IRT IO data. The reserved bandwidth ensures that the IRT data can be transmitted at reserved, synchronized intervals whilst remaining uninfluenced even by other greater network loads (e.g. TCP/IP communication or additional real time communication). The "high flexibility" enables simple planning and expansion of the system. A topological configuration is not required. Refer to page 117.
MAC address	The MAC address is an identifier unique wordlwide consisting of two parts: the first 3 bytes are the manufacturer ID and are provided by IEE standard autority; the last three bytes represent a consecutive number of the manufacturer. Refer to page 33.
Module	Modules are user defined components that plug into slots. Modules can be real or virtual.
NRT	Non Real Time - The non Real Time PROFINET IO Channel. Configuration and diagnostic messages are transferred over the NRT Channel.
Organization block	A range of organization blocks (OB) are designed to execute the user program. OBs are the interface interface between the user program and the operating system of a CPU. They permit event-controlled processing of special program components within the user program.The order in which the user program is executed is defined in the organization blocks.
Profile	Profiles define application-specific functionality to ensure the openness of PROFIBUS and PROFINET is utilized consistently. PI Profiles can cover simple devices such as encoders by defining how signals are used and how they are physically connected. However, profiles are increasingly covered more complex systems or requirements. Profiles such as PROFIdrive and PROFIsafe deliver active functionality as well. An advanced profile covering active power management for end devices like lasers and robots is now under development with the aim of bringing significant reductions in energy consumption for the automotive industry. Profiles guarantee quicker system design and they support faster device

	interchange, promoting competition amongst vendors, increased choice for users and full interoperability.
Provider Status	The Status an IO device provides to an IO Controller with the data transferred to the Controller.
Proxy	A device which maps non PROFINET IO data to PROFInet.
Real-time	Real-time means that a system processes external events within a defined time. If the reaction of a system is predictable, one speaks of a deterministic system. The general requirements for real-time are therefore: deterministic response and defined response time. Refer to page 117.
RT	Real Time - The Real Time PROFINET IO Channel. I/O and Alarm Data are transferred over the RT Channel. Refer to page 117.
Slot	A group of one or more Subslots. Slots can be real or virtual.
Standard signal	The encoder profile defines a series of standard signals which are used to configure the IO data. Refer to page 70.
Submodule	A component of a module that is plugged into a subslot. A submodule is real or virtual.
Subslot	A group of one or more channels. Subslots can be real or virtual.
Sync domain	All PROFINET devices that are to be synchronized via PROFINET IO with IRT must belong to a sync domain. The sync domain consists of precisely one sync master and at least one sync slave. IO controllers and switches can hold the role of a sync master or sync slave. Other IO devices support only the role as sync slave. Refer to page 134.
System function	System functions (SFC) are integral functions in the operating system of a S7 CPU. In addition, SFCs are frequently called implicitly by SFBs. SFCs can be called by the user program like normal functions. SFCs are used to implement a number of important system functions for Profinet IO.
System function block	System function blocks (SFB) are integral functions in the operating system of a S7 CPU. SFBs can be called by the user program like normal function blocks. SFBs are used to implement a number of important system functions for Profinet IO.
TCP/IP	The Ethernet system is designed solely to carry data. It is comparable to a highway as a system for transporting goods and passengers. The data is actually transported by protocols. This is comparable to cars and commercial vehicles transporting passengers and goods on the highway. Tasks handled by the basic Transmission Control Protocol (TCP) and Internet Protocol (IP) (abbreviated to TCP/IP): 1. The sender splits the data into a sequence of packets. 2. The packets are transported over the Ethernet to the correct recipient. 3. The recipient reassembles the data packets in the

	correct order. 4. Faulty packets are sent again until the recipient acknowledges that they have been transferred successfully.
Telegram	A telegram is a rigidly defined bit stream carrying data. A telegram specifies the data length and the type of data which is sent to and from the IO controller. The encoder profile supports Standard Telegrams 81, 82, 83 and 84. Refer to page 69.
Topology	Network structure. Commonly used structures: • Line topology; • Ring topology; • Star topology; • Tree topology. Refer to page 138.
Transmission rate	Data transfer rate (in bps).
User program	The user program contains all instructions, declarations and data for signal processing required to control a plant or a process. It is assigned to a programmable module (for example CPU) and can be structured in smaller units (blocks).

List of abbreviations

Table below contains a list of abbreviations (in alphabetical order) which may be used in this guide to describe the PROFINET IO interface. Sometimes they also refer more specifically to the S7 programming environment.

AR	Application Relation
API	Application Process Identifier
C-LS	Controller's Sign-Of-Life
CR	Communication RelationDAP
DAP	Device Access Point
DB	Data block
DO	Drive Object
DO-LS	Driver Object Sign-Of-Life
DU	Drive Unit
EO	Encoder Object
EU	Encoder Unit
FB	Function block
FC	Function
I&M	Identification & Maintenance
IRT	Isochronous Real Time Ethernet
IRT Flex	IRT "High Flexibility"
IRT Top	IRT "High Performance"
GSDML	General Station Description Markup Language
IO	Input/Output
IP	Internet Protocol
LLDP	Link Layer Discovery Protocol
LS	Sign-Of-Life
MAC	Media Access Control
MAP	Module Access Point
MLS	Master Sign-Of-Life
OB	Organization block
PAP	Parameter Access Point
PI	PROFIBUS and PROFINET International

RT	Real Time Ethernet
SFB	System function block
SFC	System function
TCP	Transmission Control Protocol
$\mathbf{T_{MAPC}}$	Master Application Cycle Time

References

1- Profile encoder. Technical Specification for PROFIBUS and PROFINET related to PROFIdrive Version 4.1 December 2008 Order No: 3.162

2- Profile Drive Technology PROFIdrive. Technical Specification for PROFIBUS and PROFINET Version 4.1 May 2006 Order No: 3.172

3- Profile Guidelines Part 1: Identification & Maintenance Functions. Guideline for PROFIBUS and PROFINET Version 1.2 October 2009 Order No: 3.502

4- Profibus Guidelines: Profibus Interconnection Technology Version V1.4 Order No: 2.142

5- Profinet Guidelines: Profinet Cabling and Interconnection Version V1.8 Order No: 2.252

1 Safety summary

1.1 Safety

- Always adhere to the professional safety and accident prevention regulations applicable to your country during device installation and operation;
- installation and maintenance operations have to be carried out by qualified personnel only, with power supply disconnected and stationary mechanical parts;
- device must be used only for the purpose appropriate to its design: use for purposes other than those for which it has been designed could result in serious personal and/or the environment damage;
- high current, voltage and moving mechanical parts can cause serious or fatal injury;
- warning ! Do not use in explosive or flammable areas;
- failure to comply with these precautions or with specific warnings elsewhere in this manual violates safety standards of design, manufacture, and intended use of the equipment;
- Lika Electronic assumes no liability for the customer's failure to comply with these requirements.

1.2 Electrical safety

- Turn OFF power supply before connecting the device;
- connect according to explanation in the "Electrical connections" section on page 28;
- in compliance with 2014/30/EU norm on electromagnetic compatibility, following precautions must be taken:

  - before handling and installing the equipment, discharge electrical charge from your body and tools which may come in touch with the device;
  - power supply must be stabilized without noise; install EMC filters on device power supply if needed;
  - always use shielded cables (twisted pair cables whenever possible);
  - avoid cables runs longer than necessary;
  - avoid running the signal cable near high voltage power cables;
  - mount the device as far as possible from any capacitive or inductive noise source; shield the device from noise source if needed;
  - to guarantee a correct working of the device, avoid using strong magnets on or near by the unit;
  - minimize noise by connecting the shield and/or the connector housing and/or the frame to ground. Make sure that ground is not affected by

noise. The connection point to ground can be situated both on the device side and on user's side. The best solution to minimize the interference must be carried out by the user. Provide the ground connection as close as possible to the encoder. We suggest using the ground point provided in the cap, use one TCEI M3 x 6 cylindrical head screw with two tooth lock washers.

1.3 Mechanical safety

- Install the device following strictly the information in the "Mechanical installation" section on page 25;
- mechanical installation has to be carried out with power supply disconnected and stationary mechanical parts;
- do not disassemble the unit unless otherwise indicated in the document;
- do not tool the unit unless otherwise indicated in the document;
- delicate electronic equipment: handle with care; do not subject the device to knocks or shocks;
- respect the environmental characteristics declared by manufacturer.

2 Identification

Device can be identified through the **order code**, the **serial number** and the **MAC address** printed on the label applied to its body. Information is listed in the delivery document too. Please always quote the order code, the serial number and the MAC address when reaching Lika Electronic for purchasing spare parts or needing assistance. For any information on the technical characteristics of the product refer to the technical catalogue.

Warning: devices having order code ending with "/Sxxx" may have mechanical and electrical characteristics different from standard and be supplied with additional documentation for special connections (Technical info).

3 Mechanical installation

WARNING
Installation and maintenance operations have to be carried out by qualified personnel only, with power supply disconnected and mechanical parts compulsorily in stop.

3.1 Overall dimensions

(values are expressed in mm)

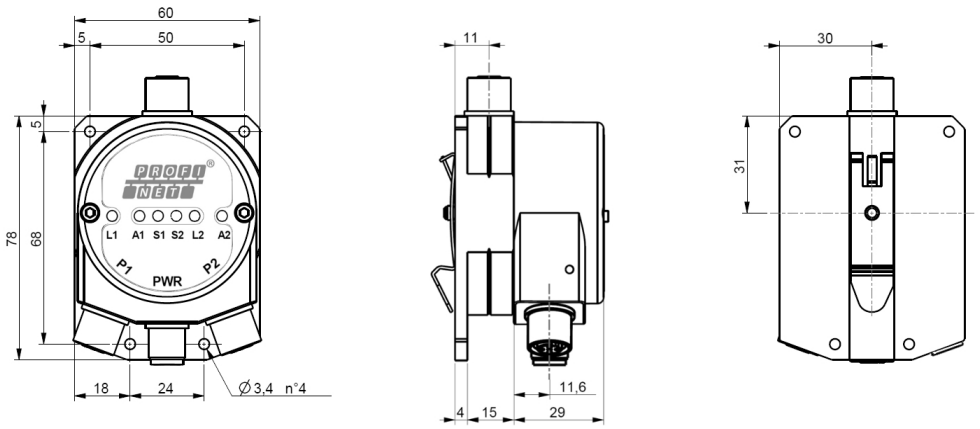


Figure 1 - Overall dimensions

3.2 Installation on panel (Figure 2)

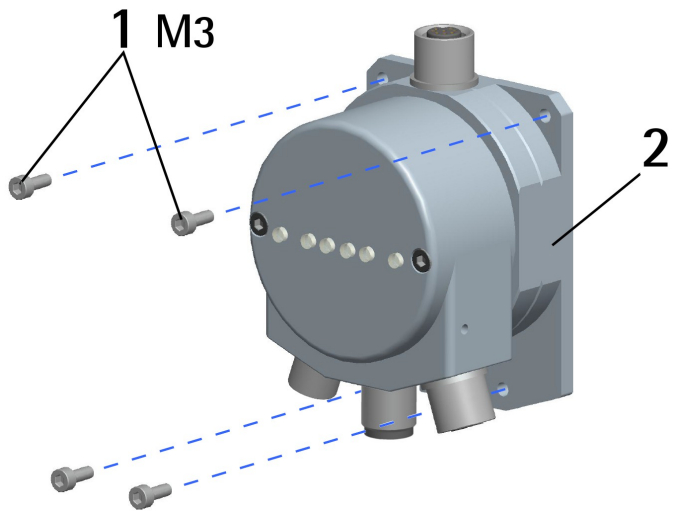


Figure 2 - Installation on panel

The unit is designed for installation on the even surface of a panel.
The back flange **2** is fitted with four holes for inserting the fixing screws **1**. Tighten the four fixing screws **1** until the unit is properly fastened to the support. Use **four M3 8 mm min. long cylinder head screws**. The recommended tightening torque is **1.1 Nm**.

3.3 Installation with DIN rail clip (Figure 3)

The unit can be installed on DIN profiles inside a rack. A clip **3** for direct fitting on DIN TS35 rails is supplied for free. It has to be fixed on the back of the flange **2** by means of the provided screw **4**.

WARNING
To mount the clip **3** you need to remove the cap **5** and drill a hole **A** in the back flange **2**. Delicate electronic circuits and wirings are located inside the cap **5**. Thus this operation has to be accomplished by skilled personnel only. Please pay careful attention and observe great precaution when carrying out this operation.

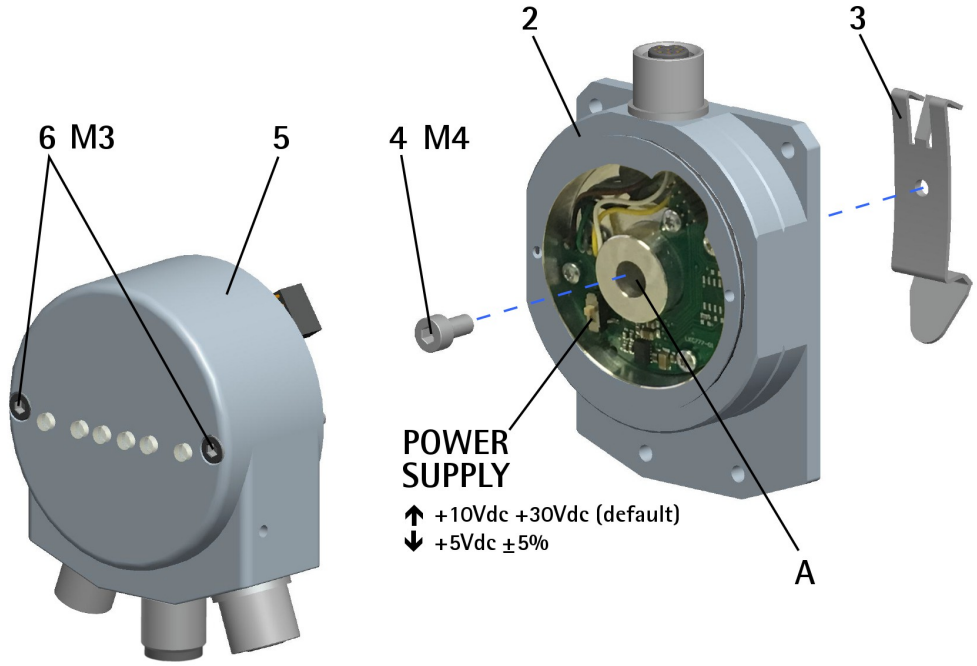


Figure 3 - Installation with DIN rail clip

- Loosen the two M3 screws **6** that fasten the cap **5** to the back flange **2**;
- open the cap **5** and separate it from the flange **2**; please pay attention to the internal wirings and connectors;
- drill a 4.5 mm diameter hole **A** in the flange **2**; use the notch in the inside of the flange **2** to guide the drill bit;

WARNING
Carefully remove the scrap material after drilling.

- mount the clip **3** on the back of the flange **2** and fix it by means of the provided M4 x 8 screw **4**; it has to be screwed on the inner side of the flange **2**;
- replace the cap **5** and fix it by means of the screws **6**.

4 Electrical connections

WARNING
Installation, electrical connection and maintenance operations must be carried out by qualified personnel only, with power supply disconnected. Mechanical components must be in stop.

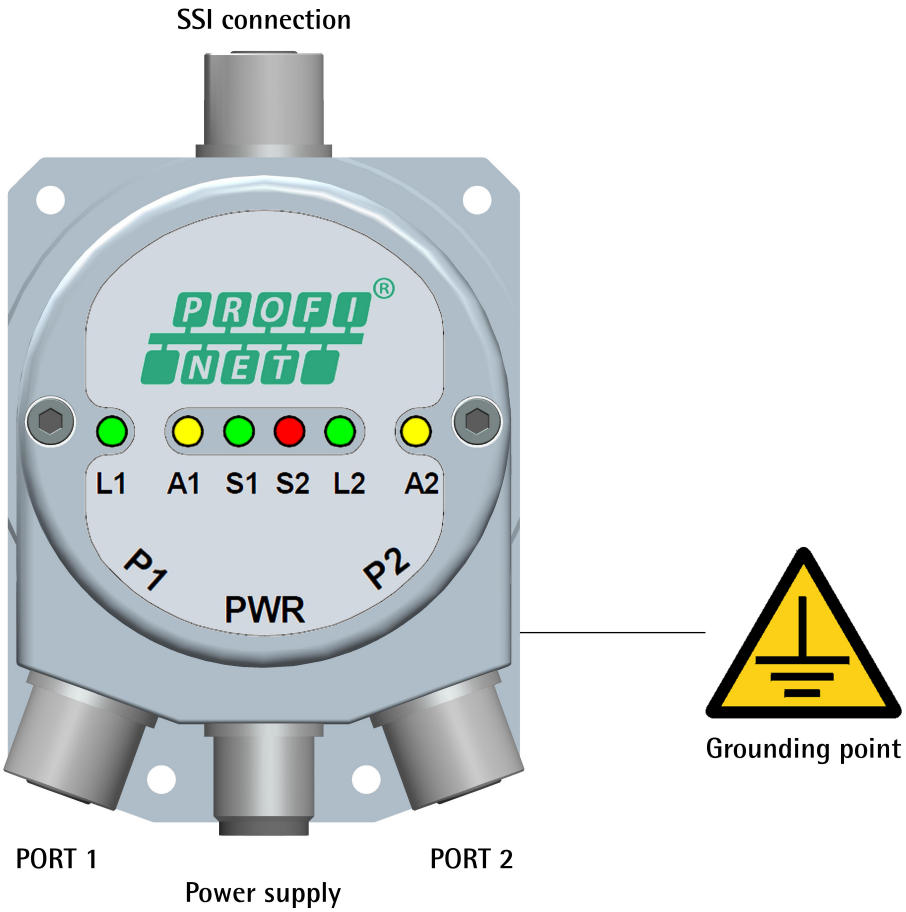
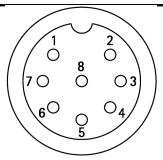


Figure 4 - Connectors and LEDs

4.1 SSI connector (Figure 4)

The connection cap is fitted with one M12 8-pin female connector to network the IF55 gateway and the SSI encoder.

M12 8-pin	SSI connection
(frontal side)	 A coding female

Pin	Description
1	0Vdc power supply
2	+Vdc power supply *
3	Clock IN +
4	Clock IN -
5	Data OUT +
6	Data OUT -
7	not connected
8	not connected

* The power supply voltage level must be set through the POWER SUPPLY DIP switch located inside the enclosure of the converter, see the following section.

WARNING
The max. length of the SSI cable must not exceed 30 m / 98.425 ft.

4.1.1 POWER SUPPLY DIP switch (Figure 5)

WARNING
Power supply must be turned off before performing this operation!

The power supply voltage level to be provided to the connected encoder must be set through the POWER SUPPLY DIP switch located inside the enclosure of the converter. It must be according to the power supply voltage level required

by the connected SSI encoder. To access the POWER SUPPLY DIP switch refer to the following section.

Set the POWER SUPPLY DIP switch to UP position to provide +10Vdc +30Vdc power supply voltage level to the encoder (default setting); set the POWER SUPPLY DIP switch to DOWN position to provide +5Vdc ±5% power supply voltage level to the encoder.

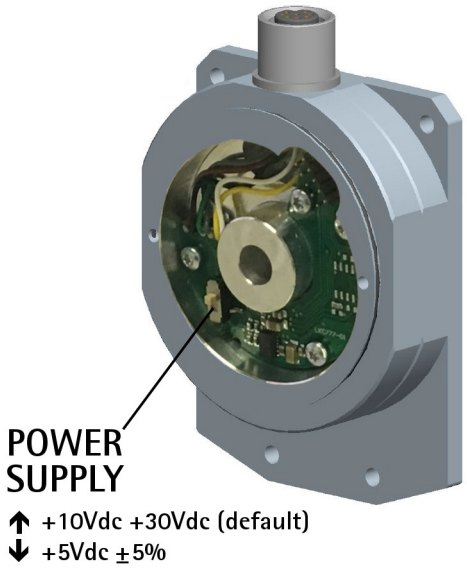


Figure 5 - POWER SUPPLY DIP switch

4.1.2 Connection cap of the converter (Figure 6)

WARNING
Do not remove or mount the connection cap with power supply switched ON. Damage may be caused to internal components.

The DIP switch meant to set the power supply of the connected SSI encoder is located inside the converter connection cap. Thus you must remove the connection cap to access it.

NOTE
Be careful not to damage the internal components when you perform this operation.

To remove the connection cap loosen the two M3 screws **1** (Figure 6). Please be careful with the internal connector.
Always replace the connection cap at the end of the operation. Take care in re-connecting the internal connector. Tighten the screws **1** using a tightening torque of approx. 2.5 Nm.

WARNING
You are required to check that the converter back flange and the connection cap are at the same potential before replacing the connection cap!

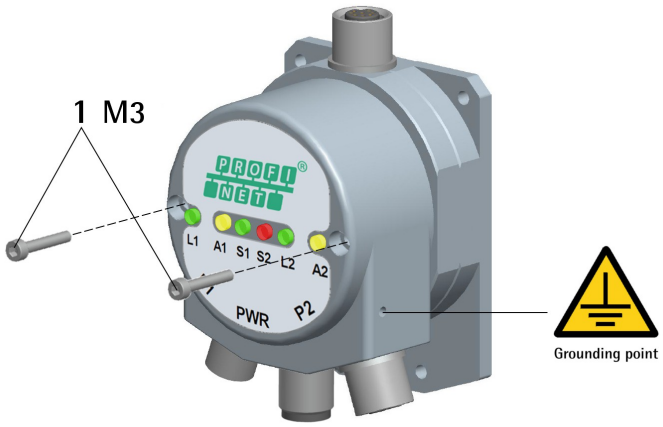
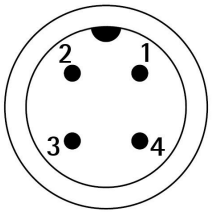


Figure 6 - Removing the connection cap

4.2 PWR Power supply connector (Figure 4)

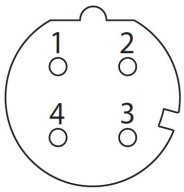
M12 4-pin male connector with A coding is used for power supply.



Description	Pin
+10Vdc +30Vdc	1
n.c.	2
0Vdc	3
n.c.	4

4.3 P1 Port 1 and P2 Port 2 connectors (Figure 4)

Two M12 4-pin female connectors with D coding are used for Ethernet connection through port 1 and port 2.



Description	Pin
Tx Data +	1
Rx Data +	2
Tx Data -	3
Rx Data -	4

The ports are equal and interchangeable - if only one connection is required, either port can be used. The Ethernet interface supports 100 Mbit/s, full-duplex operation.

4.4 Network configuration: cables, hubs, switches - Recommendations

PROFINET is based on a 100 Mbps, full-duplex Ethernet network. Faster communication is also possible on all transmission sections (e.g., between switches, PC systems, or camera systems).
Using Ethernet several topologies of connection are supported by Profinet networks: line, tree, daisy-chain, star, ... Furthermore Profinet networks can be configured in almost any topology in the same structure.
The connection of PROFINET IO field devices occurs exclusively with switches as network components. Switches typically integrated in the field device are used for this (with 2 ports assigned). PROFINET-suitable switches must support "autonegotiation" (negotiating of transmission parameters) and "autocrossover" (autonomous crossing of send and receive lines).
Cables and connectors comply with the Profinet specifications. The cabling guide defines for all Conformance Classes a 2-pair cable according to IEC 61784-5- 3.
Standard Profinet cables commercially available can be used.
The maximum segment length for electrical data transmission with copper cables between two nodes (field devices or switches) is 100 m. The copper cables are designed uniformly in AWG 22. The installation guide defines different cable types, whose range has been optimally adapted to general requirements for industry. Sufficient system reserves allow industry-compatible installation with no limitation on transmission distance.
The PROFINET cables conform to the cable types used in industry:

- PROFINET Type A: Standard permanently routed cable, no movement after installation
- PROFINET Type B: Standard flexible cable, occasional movement or vibration

- PROFINET Type C: Special applications: for example, highly-flexible, constant movement (trailing cable or torsion)

For complete information please refer to IEC 61918, IEC 61784-5-13 and IEC 61076-2-101.
To increase noise immunity only S/FTP or SF/FTP cables must be used (CAT-5).
The maximum cable length (100 meters) predefined by Ethernet 100Base-TX must be compulsorily fulfilled.
Regarding wiring and EMC measures, the IEC 61918 and IEC 61784-5-13 must be considered.

For a complete list of the available cordsets and connection kits please refer to the product datasheet ("Accessories" list).

4.5 MAC address and IP address

The unit can be identified in the network through the **MAC address** and the **IP address**. MAC address has to be intended as a permanent and globally unique identifier assigned to the unit for communication on the physical layer; while the IP address is the name of the unit in a network using the Internet protocol. MAC address is 6-byte long and cannot be modified. It consists of two parts, numbers are expressed in hexadecimal notation: the first three bytes are used to identify the manufacturer (OUI, namely Organizationally Unique Identifier), while the last three bytes are the specific identifier of the unit. The MAC address can be found on the label applied to the encoder. The IP address (and the subnet mask) must be assigned by the user to each interface of the unit to be connected in the network. For additional information on the MAC address, the IP address and the device name refer to the "5.3 Network and communication settings" section on page 49.

4.6 Line termination

Profinet network needs no line termination because the line is terminated automatically; in fact every Slave is able to detect the presence of the downstream Slaves.

4.7 Ground connection (Figure 4)

To minimize noise connect properly the shield and/or the connector housing and/or the frame to ground. Connect properly the cable shield to ground on

user's side. Lika's EC- pre-assembled cables are fitted with shield connection to the connector ring nut in order to allow grounding through the body of the device. Lika's E- connectors have a plastic gland, thus grounding is not possible. If metal connectors are used, connect the cable shield properly as recommended by the manufacturer. Anyway make sure that ground is not affected by noise. It is recommended to provide the ground connection as close as possible to the device. We suggest using the ground point provided in the cap (see Figure 4, use 1 TCEI M3 x 6 cylindrical head screw with 2 tooth lock washers).

4.8 Diagnostic LEDs (Figure 4)

Six LEDs located in the cap of the device (see Figure 4) are meant to show visually the operating or fault status of the encoder and the Profinet interface. The meaning of each LED is explained in the following tables.

LED	Description
L1 (green)	Link 1: link to another Ethernet component through port 1
A1 (yellow)	Activity 1: incoming and outgoing traffic through port 1
S1 (green)	Status 1: see table below
S2 (red)	Status 2: see table below
L2 (green)	Link 2: link to another Ethernet component through port 2
A2 (yellow)	Activity 2: incoming and outgoing traffic through port 2

S1 Status 1 green	S2 Status 2 red	Meaning	Cause
OFF	**OFF**	No power	• Unit switched off • Cable disconnected
ON	**ON**	No connection to another device Criteria: no data exchange	• Bus disconnected • Master not available or switched off
ON	**Blinking at 1 Hz**	Parametrization fault, no data exchange Criteria: data exchange is correct but the encoder does not switch to the data exchange mode	• Slave not configured yet or wrong configuration • A wrong address has been assigned to the unit • Actual configuration of the slave differs from the nominal configuration
OFF	**ON**	System failure	Diagnostic data exists, slave in data exchange mode

ON	**OFF**	Data exchange, the encoder is working properly	Correct operation
Blinking at 1 Hz	**OFF**	Flash memory upgrade process is active (see on page 164)	The user is upgrading the flash memory
OFF	**Blinking at 1 Hz**	Flash memory upgrade process failed (see on page 164)	• Cable disconnected • Power switched off • Internal error • Flash memory damaged • Upgrade process aborted before completion

5 Getting started

5.1 Quick start information

The following instructions allow the operator to quickly and safely set up the device in a standard operational mode.
For complete and detailed information please read the mentioned pages thoroughly.

- Mechanically install the device, see on page 25 ff;
- execute the electrical and network connections, see on page 28 ff;
- switch on the +10Vdc +30Vdc power supply;
- install the GSDML file, see on page 52 ff;
- add the required DAP and telegram 81 in the PROFINET-IO system, see on page 55 ff;
- set the device name, see on page 57 ff;
- if required[1], set the IP address and the subnet mask to the node see on page 57 ff; the default address set by Lika is **0.0.0.0**;
- to set the parameters, enter the **Module parameters** page, see on page 62 ff; in this page it is possible, for example, to set the singleturn resolution or the total resolution, to enable the scaling function or to change the counting direction; after entering new values, you must download the parameters to the device; the complete list of the default parameters is available on page 175; in particular you need to set the following parameters:
  - set the output code used to arrange the absolute information next to the **Code Format** parameter, see on page 91;
  - set the protocol used to arrange the absolute information next to the **Alignment** parameter, see on page 92;
  - set the number of SSI clocks next to the **Number of clocks** parameter, see on page 93.

1 Actually the manual setting of the IP address is not mandatory. If the **Set IP address in the project** check box in the **Properties** dialog window is not selected, the IP address stored in the encoder memory is used (please note that the encoder address set by default is 0.0.0.0, in compliance with Profinet specifications, it does not enable the device to go online). Otherwise, if the check box is selected, the controller automatically suggests an IP address to the device (it can be changed according to needs).

NOTE

Please consider that if the **Bypass** parameter (see on page 92) is set to "0" = disabled, the position value read by the encoder can be processed according to needs, so the user can scale the value, set a preset, and change the counting direction. On the contrary, if the **Bypass** parameter (see on page 92) is set to "1" = enabled, the information from the encoder is transmitted "as it is" and not processed in any way. The preset, scaling and counting direction functions -even if set and enabled- are ignored; also the output code setting is ignored. If, for example, the user sets a preset while the bypass mode is enabled, the value is accepted, but not activated. As soon as the bypass mode is disabled, the preset, scaling and counting direction functions -if set and enabled- become active and the position value will be accordingly.

5.1.1 Setting the physical resolution

- If you want to use the physical resolution of the encoder, please check that the **Scaling function control** parameter is disabled (="0"), see on page 89; see also the **Class 4 functionality** parameter on page 88;
- if you connected an SSI linear encoder (DAP 1), set its physical resolution next to **Physical pulse resolution [nm]** and **Physical Total resolution [bit]** parameters, see on page 95 ff;
- if you connected an SSI rotary encoder (DAP 2), set its physical resolution next to **Physical singleturn resolution [bit]** and **Physical multiturn resolution [bit]** parameters, see on page 105 ff.

5.1.2 Setting the custom resolution

- If you want to use a custom resolution, please check that the **Scaling function control** parameter is enabled (="1"), see on page 89; this parameter is active only if the **Class 4 functionality** parameter is enabled (="1"), see on page 88;
- to enable the scaling function or change the counting direction or execute the preset, the **Class 4 functionality** parameter must be enabled (="1"), see on page 88;
- if you connected an SSI linear encoder (DAP 1), set the custom resolution next to **Programmable pulse resolution [nm]** and **Programmable Total resolution [pulse]** parameters, see on page 95 ff;
- if you connected an SSI rotary encoder (DAP 2), set the custom resolution next to **Programmable pulse/rev** and **Programmable total measuring range (pulse)** parameters, see on page 106 ff.

5.1.3 Reading the position

- To read the value of the absolute position use the variable tables and the **G1_XIST1** parameter on page 72.

5.1.4 Setting and executing the preset

- To set and execute the preset proceed as follows:
  - check that the **Control by PLC** bit 10 of the **STW2_ENC** control word is ="1", see on page 74;
  - check that the **Class 4 functionality** parameter is enabled (="1"), see on page 88;
  - check that the **G1_XIST1 Preset control** parameter is enabled (="0"), see on page 89;
  - set the preset value by means of an acyclic data exchange, see the **P65000 – Preset value** parameter on page 83; or enter the **Change the preset value** page in the web server, see on page 171;
  - execute the preset by forcing high the **Request set/shift of home position** bit 12 in the **G1_STW** control word, see on page 76;
  - the encoder replies by forcing high the **Set/shift of home position executed** bit 12 in the **G1_ZSW** status word, see on page 78;
  - the Master must set back to 0 the **Request set/shift of home position** bit 12 in the **G1_STW** control word, see on page 76;
  - the **Set/shift of home position executed** bit 12 in the **G1_ZSW** status word is set back to 0, see on page 78; see the diagram on page 155.

To save the set parameters, please use the **P971 – Transfer to non volatile memory** PROFIdrive parameter (="1"), see on page 81.

5.1.5 Connecting a rotary encoder

EXAMPLE 1

We need to connect the **MM36 12/8192 BB** rotary encoder.

The main features of the rotary encoder are:
Singleturn Resolution: **12 bits = 4,096 cpr** ("12", see the order code in the product datasheet).
Multiturn Resolution: **8,192 rev. = 13 bits** ("8192", see the order code in the product datasheet).

Output code: **Binary code** (-BB-, see the order code in the product datasheet).
SSI protocol: **25-bit "LSB Right Aligned" protocol** (-BB-, see the order code in the product datasheet).

Code Format = 0 = Binary code
Alignment = 0 = 25-bit "LSB Right Aligned" protocol
Number of clocks = 25
Physical singleturn resolution [bit] = 12 (12 bits = 4,096 cpr)
Physical multiturn resolution [bit] = 13 (13 bits = 8,192 revolutions)

If you want to use the physical resolution:
Scaling function control = 0

If you need a custom resolution (Class 4 devices only):
Scaling function control = 1
Now set the resolution you need for your application next to the **Programmable pulse/rev** and **Programmable total measuring range (pulse)** parameters.

EXAMPLE 2
We need to connect the **AS58 13/BB** rotary encoder.

The main features of the rotary encoder are:
Singleturn Resolution: **13 bits = 8,192 cpr** ("13", see the order code in the product datasheet).
Output code: **Binary code** (-B..., see the order code in the product datasheet).
SSI protocol: **13-bit "LSB Right Aligned" protocol** (...B-, see the order code in the product datasheet).

Code Format = 0 = Binary code
Alignment = 0 = 13-bit "LSB Right Aligned" protocol
Number of clocks = 13
Physical singleturn resolution [bit] = 13 (13 bits = 8,192 cpr)
Physical multiturn resolution [bit] = 0 (2^0 bits = 1 revolution)

If you want to use the physical resolution:
Scaling function control = 0

If you need a custom resolution (Class 4 devices only):
Scaling function control = 1

Now set the resolution you need for your application next to the **Programmable pulse/rev** and **Programmable total measuring range (pulse)** parameters.

EXAMPLE 3

We need to connect the **AM58 13/4096 GA** rotary encoder.

The main features of the rotary encoder are:
Singleturn Resolution: **13 bits = 8,192 cpr** ("13", see the order code in the product datasheet).
Multiturn Resolution: **4,096 rev. = 12 bits** ("4096", see the order code in the product datasheet).
Output code: **Gray code** (G...-, see the order code in the product datasheet).
SSI protocol: **25-bit "LSB Right Aligned" protocol** (...A-, see the order code in the product datasheet).

Code Format = 1 = Gray code
Alignment = 0 = 25-bit "LSB Right Aligned" protocol
Number of clocks = 25
Physical singleturn resolution [bit] = 13 (13 bits = 8,192 cpr)
Physical multiturn resolution [bit] = 12 (12 bits = 4,096 revolutions)
If you want to use the physical resolution:
Scaling function control = 0

If you need a custom resolution (Class 4 devices only):
Scaling function control = 1
Now set the resolution you need for your application next to the **Programmable pulse/rev** and **Programmable total measuring range (pulse)** parameters.

EXAMPLE 4

We need to connect the **HM58 16/16384 GA** rotary encoder.

The main features of the rotary encoder are:
Singleturn Resolution: **16 bits = 65,536 cpr** ("16", see the order code in the product datasheet).
Multiturn Resolution: **16,384 rev. = 14 bits** ("16384", see the order code in the product datasheet).
Output code: **Gray code** (-GA-, see the order code in the product datasheet).

SSI protocol: **32-bit "LSB Right Aligned" protocol** (-GA-, see the order code in the product datasheet).

Code Format = 1 = Gray code
Alignment = 0 = 32-bit "LSB Right Aligned" protocol
Number of clocks = 32
Physical singleturn resolution [bit] = 16 (16 bits = 65,536 cpr)
Physical multiturn resolution [bit] = 14 (14 bits = 16,384 revolutions)

If you want to use the physical resolution:
Scaling function control = 0

If you need a custom resolution (Class 4 devices only):
Scaling function control = 1
Now set the resolution you need for your application next to the **Programmable pulse/rev** and **Programmable total measuring range (pulse)** parameters.

5.1.6 Connecting a linear encoder

EXAMPLE 1
We need to connect an **SMA5-GA-50** linear encoder.

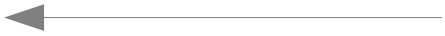
The main features of the linear encoder are:
Resolution: **0.05 mm** (-50-, see the order code in the product datasheet).
Max. measuring length: **5,050 mm** (see the "Mechanical Specifications" in the product datasheet).
Output code: **Gray code** (-GA-, see the order code in the product datasheet).
SSI protocol: **25-bit "LSB Right Aligned" protocol** (see the User's manual).

Code Format = 1 = Gray code
Alignment = 0 = 25-bit "LSB Right Aligned" protocol
Number of clocks = 25
Physical pulse resolution [nm] = 50,000 (0.05 mm resolution)
Physical Total resolution [bit] = 17 (= Max. measuring length / Resolution = 5,050 / 0.05 = 101,000 ≈ 2^{17} = 17 bits)

If you want to use the physical resolution:
Scaling function control = 0

If you need a custom resolution:
Scaling function control = 1
Programmable pulse resolution [nm] ≤ **Physical pulse resolution [nm]**, the user can set a custom measuring step
Programmable Total resolution [pulse] ≤ 131,072 (= 5,050 / 0.05 = 101,000 information; max. value 2^{17} = 131,072 dec); the user can set a custom measuring range

If you set a 0 preset along the path, when the encoder moves back and cross the zero, the value immediately after 0 will be 2^{17} - 1, i.e. 131,071 (assuming that **Programmable Total resolution [pulse]** = 131,072; **Physical Total resolution [bit]** = 2^{17} = 131,072).



...	131,069	131,070	131,071	0	1	2	...

EXAMPLE 2
We need to connect an **SMA5-GA-100** linear encoder.

The main features of the linear encoder are:
Resolution: **0.1 mm** (-100-, see the order code in the product datasheet).
Max. measuring length: **5,050 mm** (see the "Mechanical Specifications" in the product datasheet).
Output code: **Gray code** (-GA-, see the order code in the product datasheet).
SSI protocol: **25-bit "LSB Right Aligned" protocol** (see the User's manual).

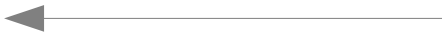
Code Format = 1 = Gray code
Alignment = 0 = 25-bit "LSB Right Aligned" protocol
Number of clocks = 25
Physical pulse resolution [nm] = 100,000 (0.1 mm resolution)
Physical Total resolution [bit] = 16 (= Max. measuring length / Resolution = 5,050 / 0.1 = 50,500 ≈ 2^{16} = 16 bits)

If you want to use the physical resolution:
Scaling function control = 0

If you need a custom resolution:
Scaling function control = 1
Programmable pulse resolution [nm] ≤ **Physical pulse resolution [nm]**, the user can set a custom measuring step

Programmable Total resolution [pulse] ≤ 65,536 (= 5,050 / 0.1 = 50,500 information; max. value 2^{16} = 65,536 dec); the user can set a custom measuring range

If you set a 0 preset along the path, when the encoder moves back and cross the zero, the value immediately after 0 will be 2^{16} - 1, i.e. 65,535 (assuming that **Programmable Total resolution [pulse]** = 65,536; **Physical Total resolution [bit]** = 2^{16} = 65,536).



...	65,533	65,534	65,535	0	1	2	...

EXAMPLE 3

We need to connect an **SMAX-BG-100** linear encoder.

The main features of the linear encoder are:
Resolution: **0.1 mm** (-100-, see the order code in the product datasheet).
Max. measuring length: **600 mm** (see the "Mechanical Specifications" in the product datasheet).
Output code: **Binary code** (-BG-, see the order code in the product datasheet).
SSI protocol: **13-bit "MSB Left Aligned" protocol** (see the User's manual).

Code Format = 0 = Binary code
Alignment = 1 = 13-bit "MSB Left Aligned" protocol
Number of clocks = 13
Physical pulse resolution [nm] = 100,000 (0.1 mm resolution)
Physical Total resolution [bit] = 13 (= Max. measuring length / Resolution = 600 / 0.1 = 6,000 ≈ 2^{13} = 13 bits)

If you want to use the physical resolution:
Scaling function control = 0

If you need a custom resolution:
Scaling function control = 1
Programmable pulse resolution [nm] ≤ **Physical pulse resolution [nm]**, the user can set a custom measuring step
Programmable Total resolution [pulse] ≤ 8,192 (= 600 / 0.1 = 6,000 information; max. value 2^{13} = 8,192 dec); the user can set a custom measuring range

If you set a 0 preset along the path, when the encoder moves back and cross the zero, the value immediately after 0 will be 2^{13} - 1, i.e. 8,191 (assuming that **Programmable Total resolution [pulse]** = 8,192; **Physical Total resolution [bit]** = 2^{13} = 8,192).

...	8,189	8,190	8,191	0	1	2	...

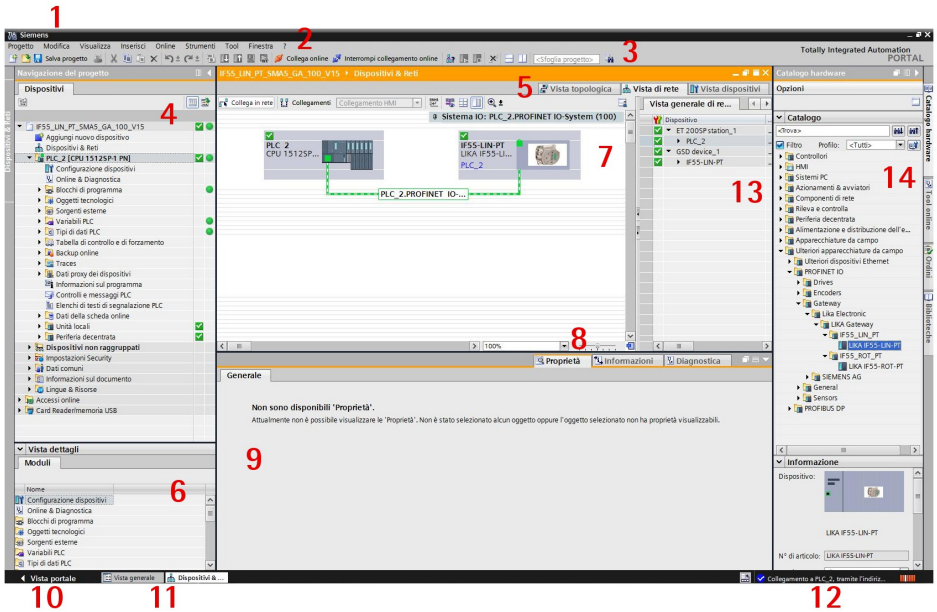
5.2 Configuring the gateway using Siemens TIA PORTAL

In this manual some screenshots are shown to explain how to install and configure the encoder in a supervisor. In the specific example the development environment is TIA PORTAL V15 with SIEMENS PLC CPU 1512SP-1 PN. Therefore, the installation of the GSDML file, the assignment of the IP address and the device name, the configuration of the device in the network, topology, diagnostics, etc. will always refer to the aforementioned development tools. If you need to install the device using a different configuration tool, please read and follow carefully the instructions given in the documentation provided by the manufacturer.

5.2.1 About TIA Portal

TIA Portal stands for Totally Integrated Automation Portal. It is an integrated engineering framework for controllers, HMI and drives. It integrates several SIMATIC products into a single software in order to increase productivity and efficiency.
TIA portal can be used to configure both the PLC and the visualization in an homogeneous system. Data is saved in a single project. Tools for programming (STEP 7) and displaying (WinCC) are not distinct programs, but editors of a system that has access to and uses a common database. One single user interface is used to enter all functions used for displaying and programming.

5.2.2 Project overview



1. **Title bar**: the name of the project is displayed in the title bar.
2. **Menu bar**: the menu bar contains all the commands that you require for your work.
3. **Toolbar**: the toolbar provides you with buttons for commands you will use frequently. This gives you faster access to these commands.
4. **Project Tree**: using the Project Tree features gives you access to all components and project data. You can perform the following tasks in the Project Tree:
    - add new components
    - edit existing components
    - scan and modify the properties of existing components
5. **Changeover switches**: they allow to switch among the three working areas of the **Hardware and network editor**: Topology view, Network view and Device view. See point 7 for more information.
6. **Details view**: it shows certain content of the selected object in the **Overview Window** or in the **Project Tree**. This might include text lists or tags. The content of the folders is not shown, however. To display the content of the folders, use the **Project Tree** or the **Inspector Window**.
7. **Graphic Area** of the **Hardware and network editor**. The **Hardware and network editor** opens when you double-click on the **Devices and Networks** entry in the **Project Tree**. The **Hardware and network editor** is the integrated development environment for configuring, networking and assigning parameters to devices and modules. It provides maximum support for the realization of the automation project. This pane is the graphic area where the current configuration of the installed devices with information on the topology and the network can be found. The **Hardware and network editor** provides you with three views of your project. You can switch between these three views at any time depending on whether you want to produce and edit individual devices and modules, entire networks and device configurations or the topological structure of your project.
   See the **Changeover switches**, point 5: **Device view** for parametrisation and configuration of the individual devices, it allows to configure and assign both device and module parameters, see on page 48; **Network view** for graphical connections between devices, it allows to configure and assign device parameters and to network the devices with one another, see on page 48; and **Topology view** for current interconnection of Profinet devices, it allows to display and configure the Ethernet topology as well as to identify and minimize differences between the desired and actual topology, see on page 49. In the Figure above the SIEMENS PLC CPU 1512SP-1 PN is the Master device and is connected to the IF55 LIN PT gateway, i.e. the Slave device, through the PLC_2.PROFINET IO-... connection.

8. **Overview Navigation**, it allows to quickly scroll through the objects available in the **Work Area** by pressing the left button of the mouse.
9. **Inspector window**: additional information on an object selected or on actions executed are displayed in the **Inspector window**, the available properties and parameters shown for the object selected can be edited in the Inspector window using the **Properties** tab.
10. It allows to enter the **Portal view**. The Portal view provides you with a task-oriented view of the tools.
11. **Editor bar**: it displays the open editors. If you have opened a lot of editors, they are shown grouped together. You can use the Editor bar to change quickly between the open elements.
12. **Status bar with progress display**. In the status bar, you will find the progress display for processes that are currently running in the background. This also includes a progress bar that shows the progress graphically. Hover the mouse pointer over the progress bar to display a tooltip providing additional information on the active background process. You can cancel the background processes by clicking the button next to the progress bar. If no background processes are currently running, the status bar displays the last generated alarm.
13. **Table Area** of the **Hardware and network editor**: it offers a general overview of the characteristics of the Device (when **Device view** is selected), of the Network (when **Network view** is selected) and of the Topology (when **Topology view** is selected).
14. **Task Cards**: depending on the edited or selected object, task cards that allow you to perform additional actions are available. These actions include:
    - selecting objects from a library or from the hardware catalog
    - searching for and replacing objects in the project
    - dragging predefined objects to the work area

    The task cards available can be found in a bar on the right-hand side of the screen. You can collapse and reopen them at any time. Which task cards are available depends on the products installed. More complex task cards are divided into panes that you can also collapse and reopen.

    The **Hardware catalog** can be selected in the **Task Cards**; it allows to install the available components just dragging and dropping them onto the **Work Area**. Customarily the field devices that have been integrated into the TIA Portal via GSDML files are listed under **Other field devices > Profinet IO**.

5.2.3 Device view

Press the **Device view** changeover switch in the **Hardware and network editor** to enter the **Device view**.
The configuration of devices and assigning of addresses etc. is performed in the **Device view**. All devices are represented in a photo-realistic way.

- Buffering of configured hardware modules and reuse with module clipboard
- When zoomed to at least 200%, I/Os are displayed with the symbolic names / addresses
- Automatic readout of available hardware with hardware detect
- Full text search in the Hardware catalogue
- Option of filtering the Hardware catalogue to show modules that can currently be used
- All parameters and configuration data are displayed on a hierarchical and context-sensitive basis

5.2.4 Network view

Press the **Network view** changeover switch in the **Hardware and network editor** to enter the **Network view**.
The **Network view** enables the configuration of plant communication. The communication links between individual stations are displayed here graphically and very clearly.

- Combined view of all network resources and network components
- Fully graphical configuration of the individual stations
- Resources are networked by linking communication interfaces using drag & drop
- Multiple controllers, peripherals, HMI devices, SCADA stations, PC stations and drives possible in a single project
- Procedure for integrating AS-i devices identical to PROFIBUS/PROFINET
- Zoom and page navigation
- Copying/pasting entire stations, incl. configuration, or individual hardware modules

A subnet (PLC_2.PROFINET IO) is added to the operator panel. Click the subnet (PLC_2.PROFINET IO) to apply the network settings. Specify the required network settings under **Properties > Network Settings** in the **Properties** area (see point 9 on page 45). Make sure that you use the same settings throughout the entire network.

5.2.5 Topology view

Press the **Topology view** changeover switch in the **Hardware and network editor** to enter the **Topology view**.
Decentralised peripherals on Profinet are configured in the Network view. The controllers and the decentralised peripherals assigned to them can be shown graphically. During ongoing operation, however, it is not possible to see which ports are actually connected and communicating with each other.
Yet this is precisely what is often important for diagnostics. For Profinet networks, the **Topology view** enables this information to be displayed quickly and easily. An offline/online comparison identifies the communicating ports. By detecting, presenting and monitoring the physical connections between devices on Profinet, the administrator can easily monitor and maintain even complex networks.

5.3 Network and communication settings

The **MAC address** of the device is reported in the label applied to the actuator enclosure. See the following section.
The IP address and the subnet mask as well as the Profinet device name must be assigned by the user to each interface of the unit to be connected in the network. By default, before delivery the device name of the converter is set to a **blank string** and its IP address is set to **0.0.0.0**. See on page 56.

5.4 Mac address

The MAC address is an identifier unique worldwide.
The MAC-ID consists of two parts: the first 3 bytes are the manufacturer ID and are provided by IEE standard authority; the last three bytes represent a consecutive number of the manufacturer.

NOTE
The MAC address is always printed on the label for commissioning purposes.

The MAC address has the following structure:

Bit value 47 ... 24			**Bit value 23 ... 0**		
10	B9	FE	X	X	X
Company code (OUI)			**Consecutive number**		

5.5 Converter installation under TIA PORTAL environment

5.5.1 Description of the GSDML file

The functionality of a PROFINET IO device is always described in a GSDML file. This file contains all data that are relevant for engineering as well as for data exchange with the IO device.
PROFINET IO devices can be described using XML-based GSD. The description language of the GSD file, i.e. GSDML (General Station Description Markup Language) is based on international standards. As the name suggests, the GSD file is a language-independent XML file (Extensible Markup Language).

Profinet gateways from Lika Electronic are supplied with their own GSDML file **GSDML-V2.25-LIKA-0239-IF55-PT-XXXXXXXX.XML** where XXXXXXXX is the release date of the file in a 8-digit format encompassing information about year (4 digits), month (2 digits) and day (2 digits): **20180319** is a sample of a GSDML file released by Lika Electronic for Profinet converters. Enter **www.lika.biz > DISPLAYS & INTERFACES > SIGNAL CONVERTERS & INTERFACES (POSICONTROL)** to get the GSDML file.
The XML file has to be installed in the Profinet Controller.

Version structure of GSDML files
The GSDML file structure is in compliance with the ISO 15745 "Open Systems Application Integration Framework" and is oriented on the defined profile of a field device via the following model:

GSDML-	**V2.25-**	**LIKA-0239-**	**IF55-PT-**	**20180319**	**.xml**
GSD data identification	Version of GSDML scheme	Manufacturer	Name of device	Version number, format: yyyymmdd	File extension

- The version of the GSDML model used defines which scope of language a GSD file uses.
- The version date is updated, if, for example, an error is cleared or a function extended.

WARNING
Please always comply with the specifications indicated in the following table:

GSDML file version	Converter HW version	Converter SW version	User's guide version
Release 20180319	1	1.0.1	1.0
From release 20181129 to ...	1	1.0.2	1.1

5.5.2 Installing the GSDML file

To install the gateway on TIA Portal proceed as follows.

1. In the TIA Portal, select the **Options > Manage general station description files (GSD)** menu.

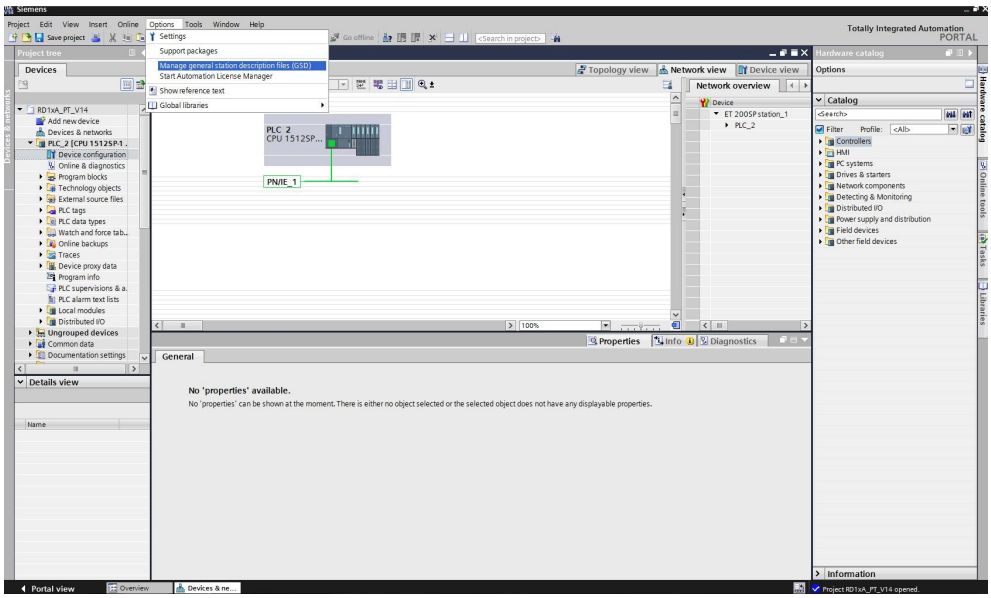


Figure 7 - Installing the GSDML file

2. In the **Manage general station description files** dialog box, select the directory containing the GSDML files.
3. Select the GSDML file specific to the device you need to install.
4. Click the **INSTALL** button.

Customarily the field devices that have been integrated into the TIA Portal via GSDML files are listed under **Other field devices > Profinet IO**, see the "5.2.2 Project overview" section on page 45.

5.5.3 Adding a node to the project

On the right side of the TIA Portal, open the **Hardware catalog** task card. The field devices integrated into the TIA Portal via the Profinet file (GSDML file) can then be found under **Other field devices\PROFINET IO\General\Lika Electronic\Lika GATEWAY**.
Select the desired device (DAP 1 and DAP 2) from the **Lika GATEWAY** directory (for instance, **LIKA IF55-LIN-PT**) and use drag-and-drop to move the item from the editing window to the **Network view**. This creates the device in the project. Detailed information on the selected device is available at the bottom of the Hardware catalog in **Information**.

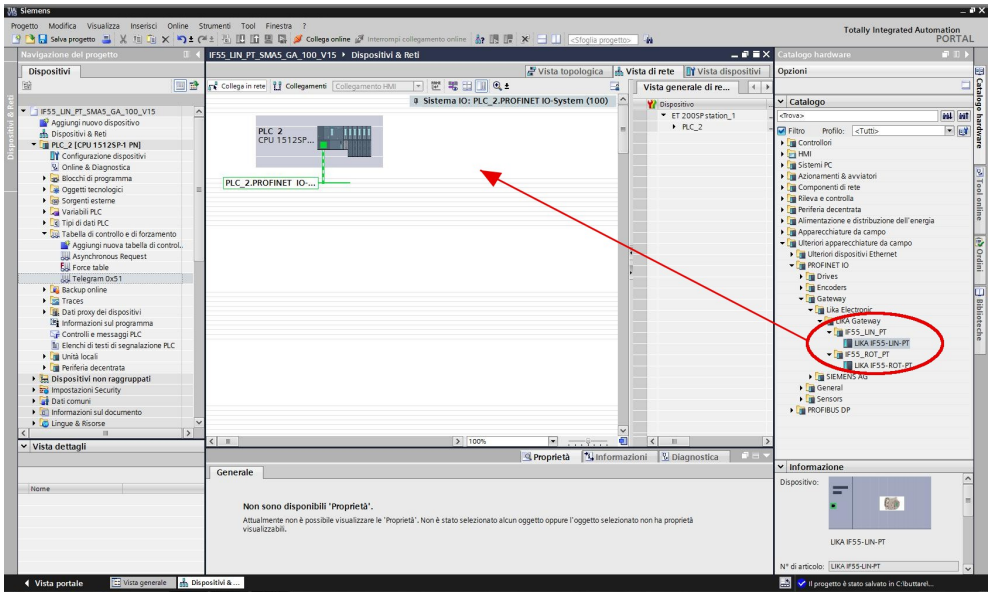


Figure 8 - Adding a node to the project

5.5.4 Establishing the bus connection

As soon as the device has been inserted into the project, the bus connection with the PLC can be established in the **Network view**.
The "**Not assigned**" information message appears in the node picture: it warns that the connection between the PLC and the Slave device is not established yet. Right-click on the message and select, through the **Select IO controller** drop-down box, the PLC the node has to be connected to. When doing so, make sure that you are in the **Network** function mode in the **Network view**.

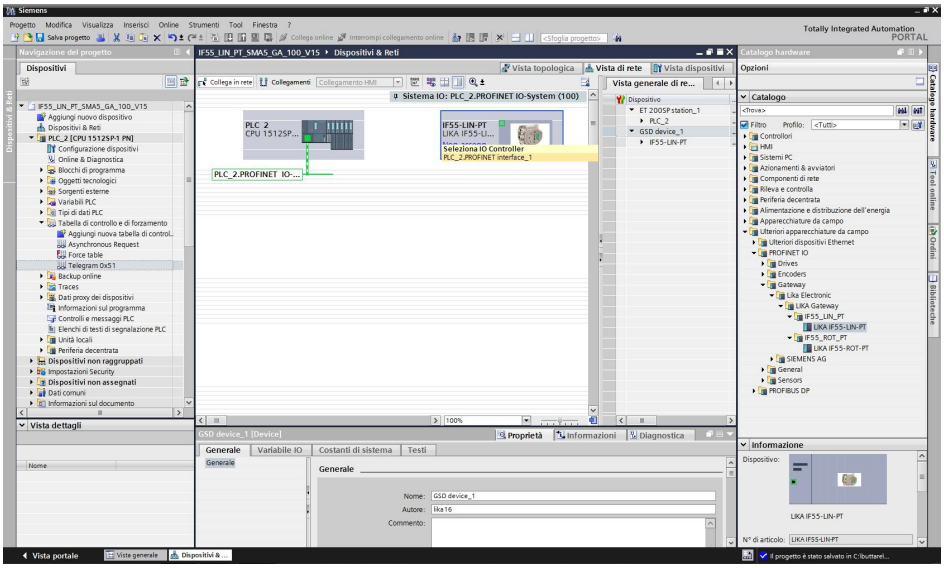


Figure 9 - Establishing the bus connection

After configuring the networking, the device is connected to the PLC via the PROFINET network.

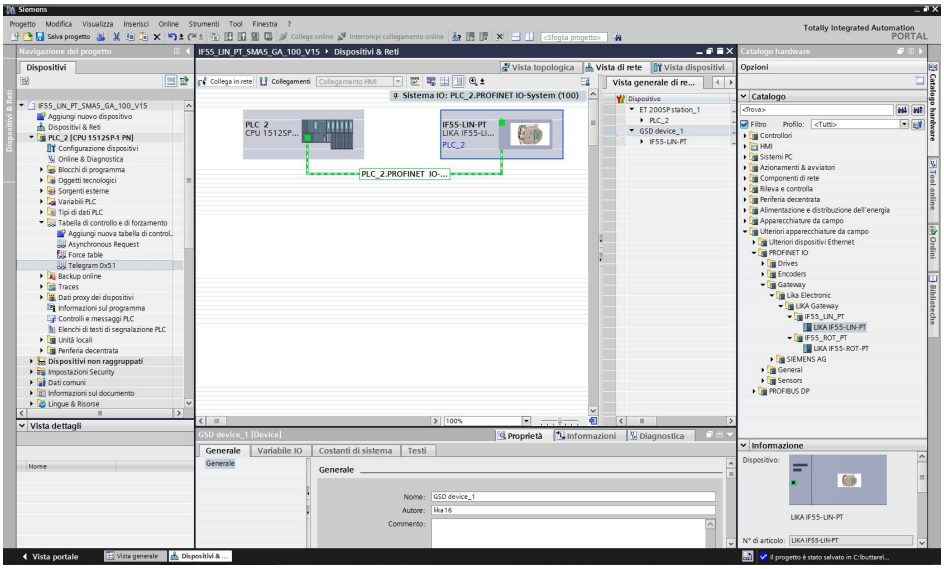


Figure 10 - Device connected

5.5.5 Inserting the Standard Telegram 81

Now we need to choose the data length and the type of data that should be sent to and from the IO controller, thus we need to install a Standard Telegram. Lika Electronic's gateways implements the Standard Telegram 81. For detailed information on the Standard Telegrams refer to the section "7.1 Telegrams" on page 69.

To install the Standard Telegram 81 select it from the **Hardware Catalog** in the **Task Cards** and drag and drop it onto the **Table Area**, as shown in the Figure.

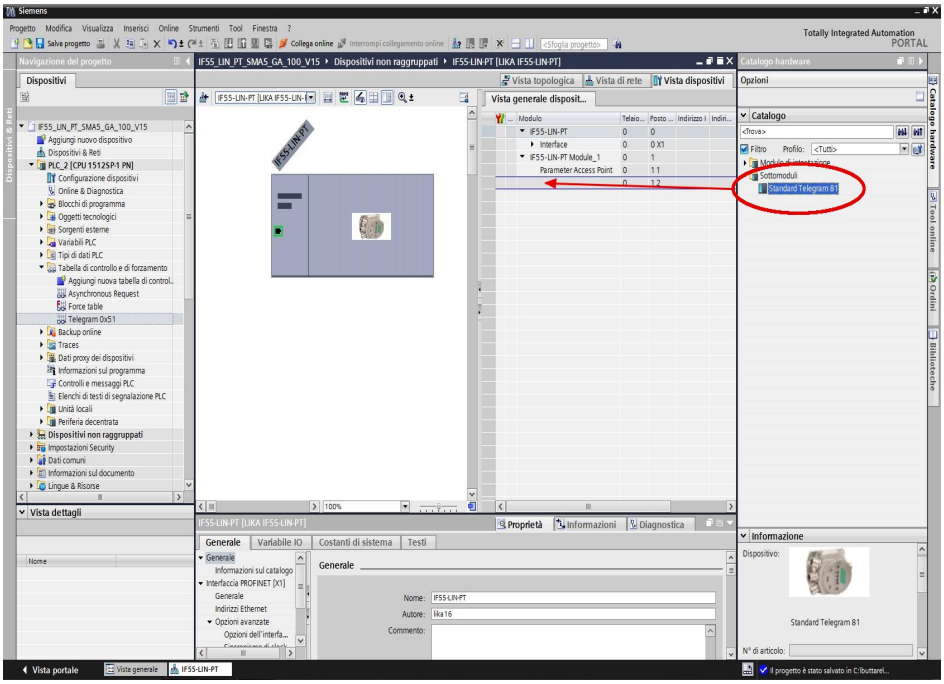


Figure 11 - Adding the Standard Telegram 81

5.5.6 Device name and IP address at delivery

In a Profinet network it is mandatory that each IO device is provided with its own Device name and IP address. By default, before delivery the device name of the converter is set to a **blank string** and its IP address is set to **0.0.0.0**.

Before the PROFINET IO controller can address a PROFINET IO device, a name has to be assigned to the PROFINET IO device. PROFINET uses this method because names are easier to use and recall than complex IP addresses. Devices on an Ethernet subnet must have unique names.

NOTE
An IO Device does not have a device name when delivered. By default, the device name of Lika's Profinet converters is set to a **blank string**.

The device names must satisfy DNS (Domain Name System) conventions:

- Names are limited to a total of 127 characters (letters, numbers, dashes or dots).
- Any component part (that is, a character string between two dots) of the device name may only be up to 63 characters long.
- Names cannot contain any special character such as umlauts, parentheses, underscores, forward or backward slashes, empty spaces, etc. The dash is the only special character allowed.
- Names must neither start nor end with the minus "-" sign.

5.5.7 Setting the device name and the IP address

As stated, to completely establish the connection, you have to assign the IP address and the Profinet device name to the Slave device. To do so, enter the **Device view** working area, select the device you need to configure in the drop-down box on the top left of the graphic area, right-click on the image of the module and select the **Properties** command from the shortcut menu.

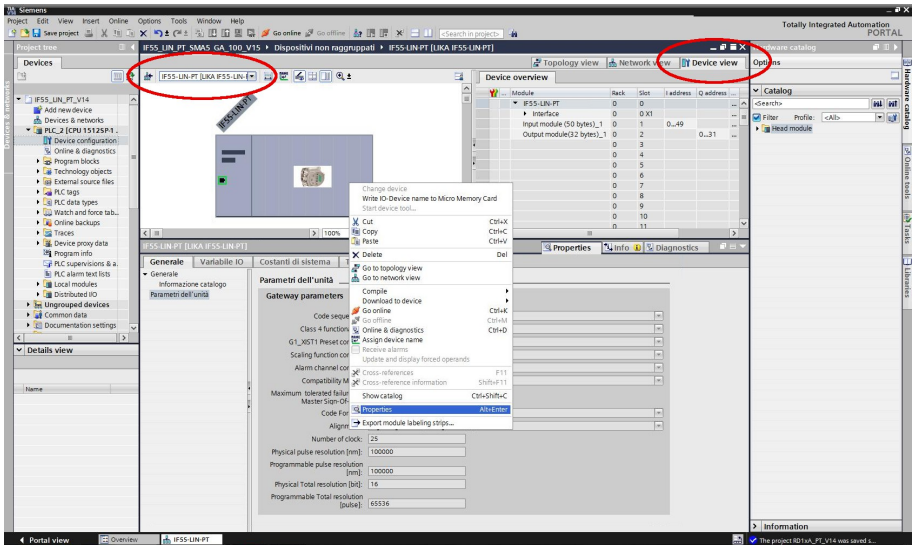


Figure 12 - Setting the device name and IP address

In the **Properties** inspector window, **General** tab, you can now use the **Ethernet addresses** menu option to set the Ethernet address (IP address, subnet mask, ...) and assign the Profinet name of the Device.

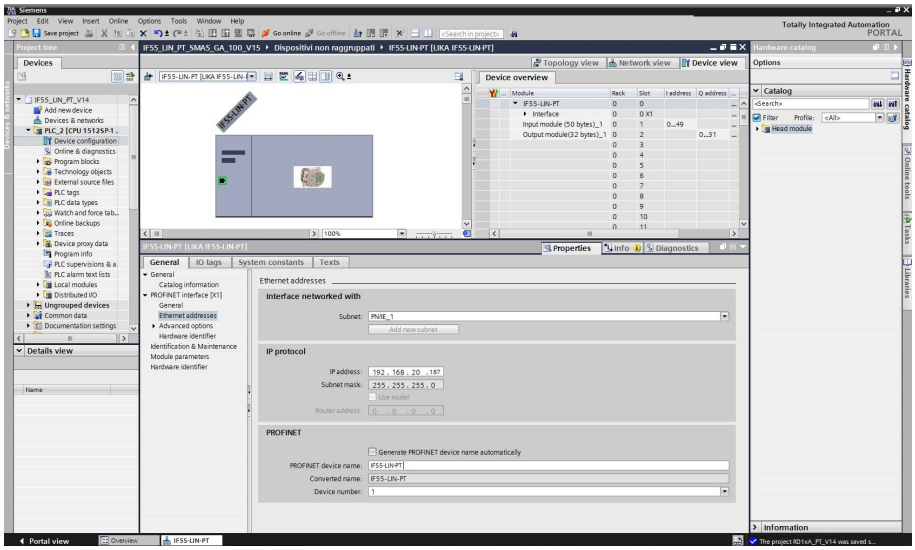


Figure 13 - Setting the device name and IP address

5.5.8 Compiling and transferring the project

After setting you must compile and then transfer the project to the device.

5.5.9 Establishing an online connection (Online mode)

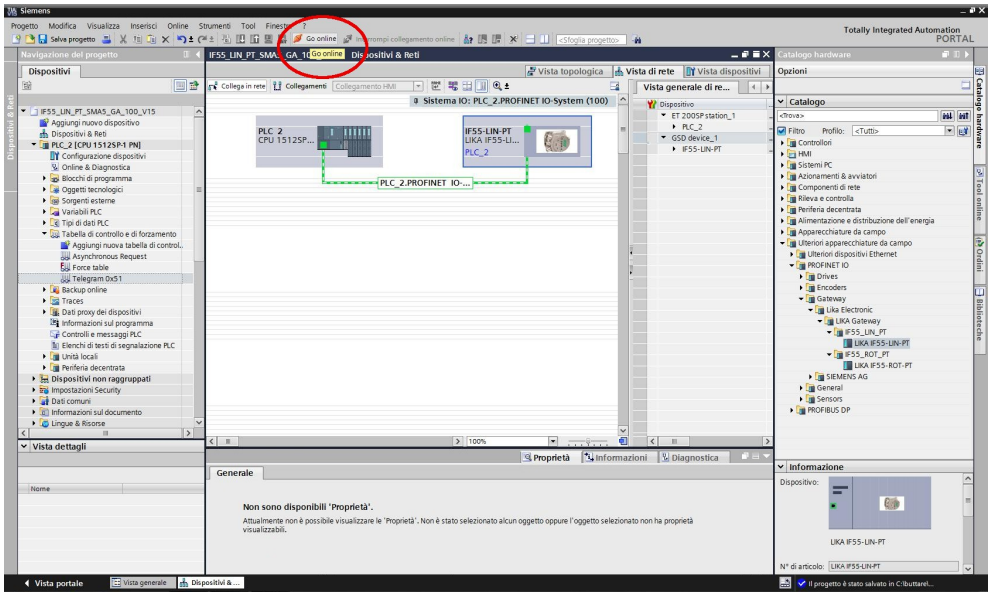


Figure 14 - Establishing an online connection

In online mode, there is an online connection between the PLC and one or more devices. An online connection between the PLC and the device is required, for example, for the following tasks:

- Using the Control Table
- Testing user programs
- Displaying and changing the operating mode of the device
- Displaying module information
- Comparing blocks
- Hardware diagnostics

Before you can establish an online connection, the PLC and the device must be physically or remotely connected.
After establishing a connection, you can use the **Online and Diagnostics view** or the **Online tools** task card to access the data on the device. The current online status of a device is indicated by an icon to the right of the device in the **Project Tree**.

To establish an online connection between the PLC (Profinet Controller) and the device (Profinet Device) proceed as follows.

- In the **Project Tree** (see point 4 in the "5.2.2 Project overview" section on page 45) mark the folder of the PLC that is configured as the Controller.
- Select the **Go online** command in the **Online** menu bar to establish an online connection to the PLC (Controller) and to the device (Device).
- If the device has already been connected online, the online connection is automatically established using the previously specified connection path.
- If there was no previous connection, the **Go online** dialog opens.
- Select the connection path:
  - select the type of interface;
  - select the interface of the PLC;
  - select the interface or the subnet for the connection.
- Click the **START SEARCH** button. Devices which can be reached by the set connection path are displayed in the **Compatible devices in target subnet**. The connection line in the graphic is displayed as solid.
- Select the device in the **Compatible devices in target subnet table** and confirm the selection with **Go online**. The online connection to the selected target device is established.

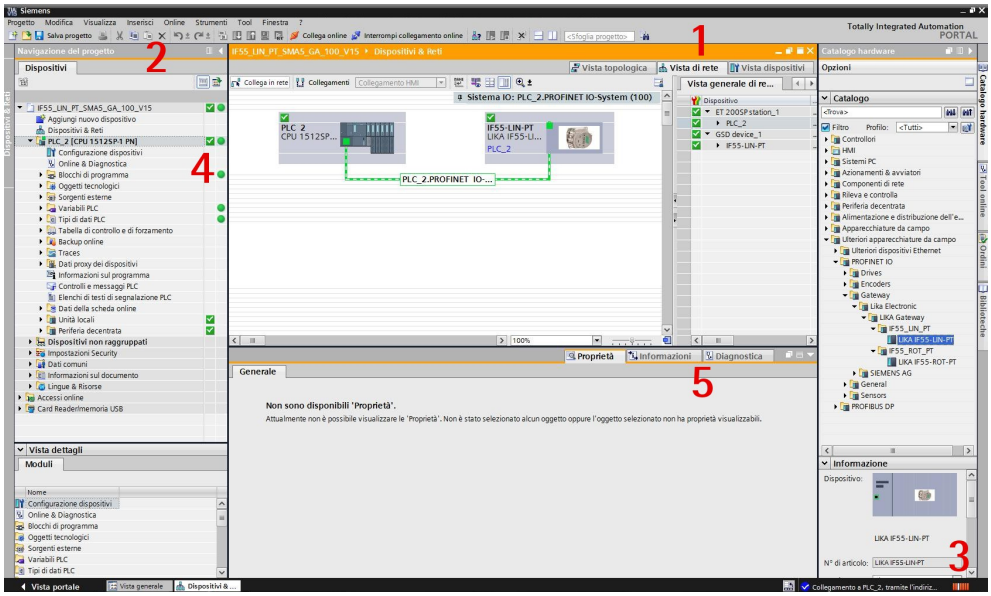


Figure 15 - Online connection established

After the online connection has been established successfully, the user interface changes (see the Figure above).

1. The title bar of the active window gets an orange background as soon as at least one of the devices currently displayed in the editor has been successfully connected online. If one or more devices are unavailable, a symbol for a broken connection appears in the title bar of the editor.
2. The title bars of inactive windows for the relevant station now have an orange line below them.
3. An orange, pulsing bar appears at the right-hand edge of the status bar. If the connection has been established but it is not working properly, an icon for an interrupted connection is displayed instead of the bar. You will find more information on the error in **Diagnostics** in the **Inspector window**.
4. Operating mode symbols or diagnostics symbols for the stations connected online and their underlying objects are shown in the **Project Tree**. A comparison of the online and offline status is also made automatically. Differences between online and offline objects are also displayed in the form of symbols.
5. The **Diagnostics > Device information** area is brought to the foreground in the **Inspector window**.

5.5.10 Closing an online connection

To close the existing online connection, follow these steps.

1. Select the device for which you want to disconnect the online connection in the **Project Tree**.
2. Select the **Go offline** command in the **Online** menu bar. The online connection is disconnected.

5.5.11 Diagnostics

Configuration of the diagnostics is integrated in the system in a user-friendly way and activated with just one click. When new hardware components are introduced, the diagnostic information is updated automatically via the engineering system (HWCN). System diagnostics outputs all relevant information on existing errors in the system. This information is packaged automatically in messages containing the following elements:

- Module
- Message text
- Message status

To access the diagnostics function please proceed as follows.

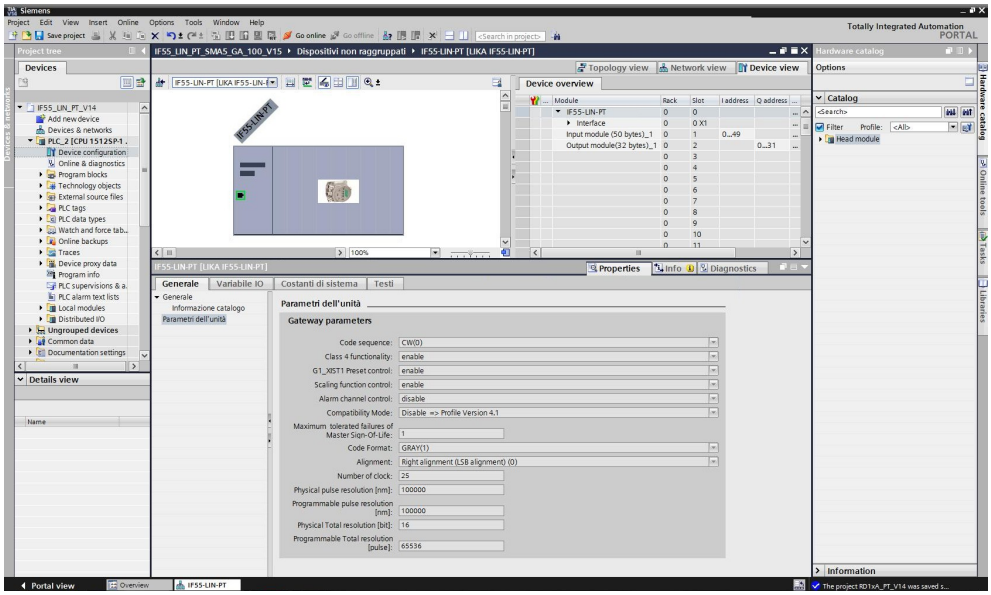
1. Right-click on the module to process.
2. Select the **Online & diagnostics** command from the shortcut menu.
3. If there is no online connection established, click the **Connect online** button in the **Diagnostics** entry.
4. The diagnostic status of the module will be displayed in the **Diagnostic status** group in the **Diagnostics** folder in the **Online and diagnostics view** of the module to be diagnosed.

The following status information is displayed in the **Diagnostic status** area:

- Status of the module as viewed by the CPU, for example:
  - Module available and OK.
  - Module defective.
    If the module experiences a fault and you have enabled the diagnostic error interrupt during configuration, the "Module defective" status is displayed.
  - Module configured, but not available.
    Example: Diagnostics data is not available because the current online configuration differs from the offline configuration.
- Detected differences between the configured and the inserted module. Provided it can be ascertained, the article number will be displayed for the set and actual type.

The scope of the displayed information depends on the selected module.

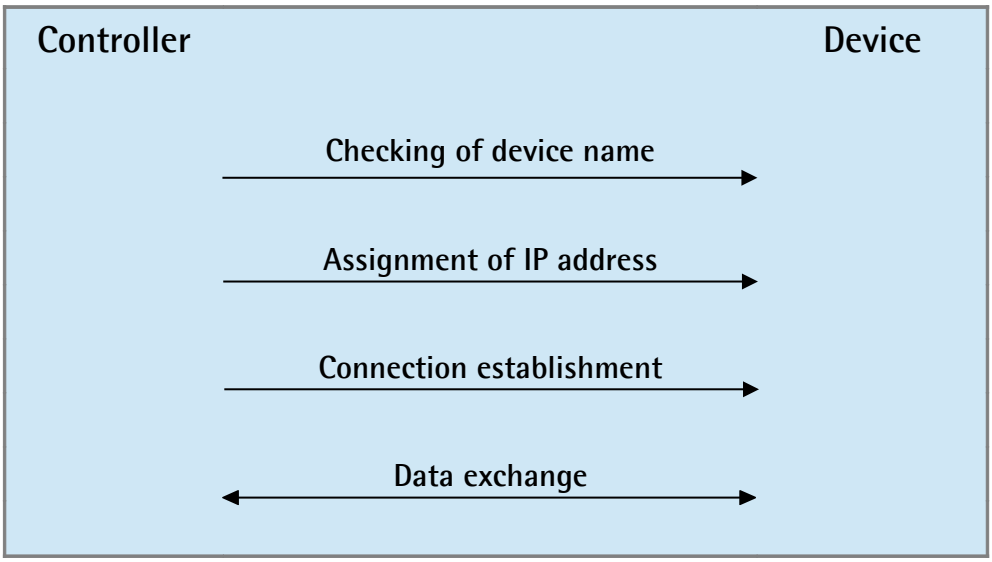
5.5.12 Module parameters



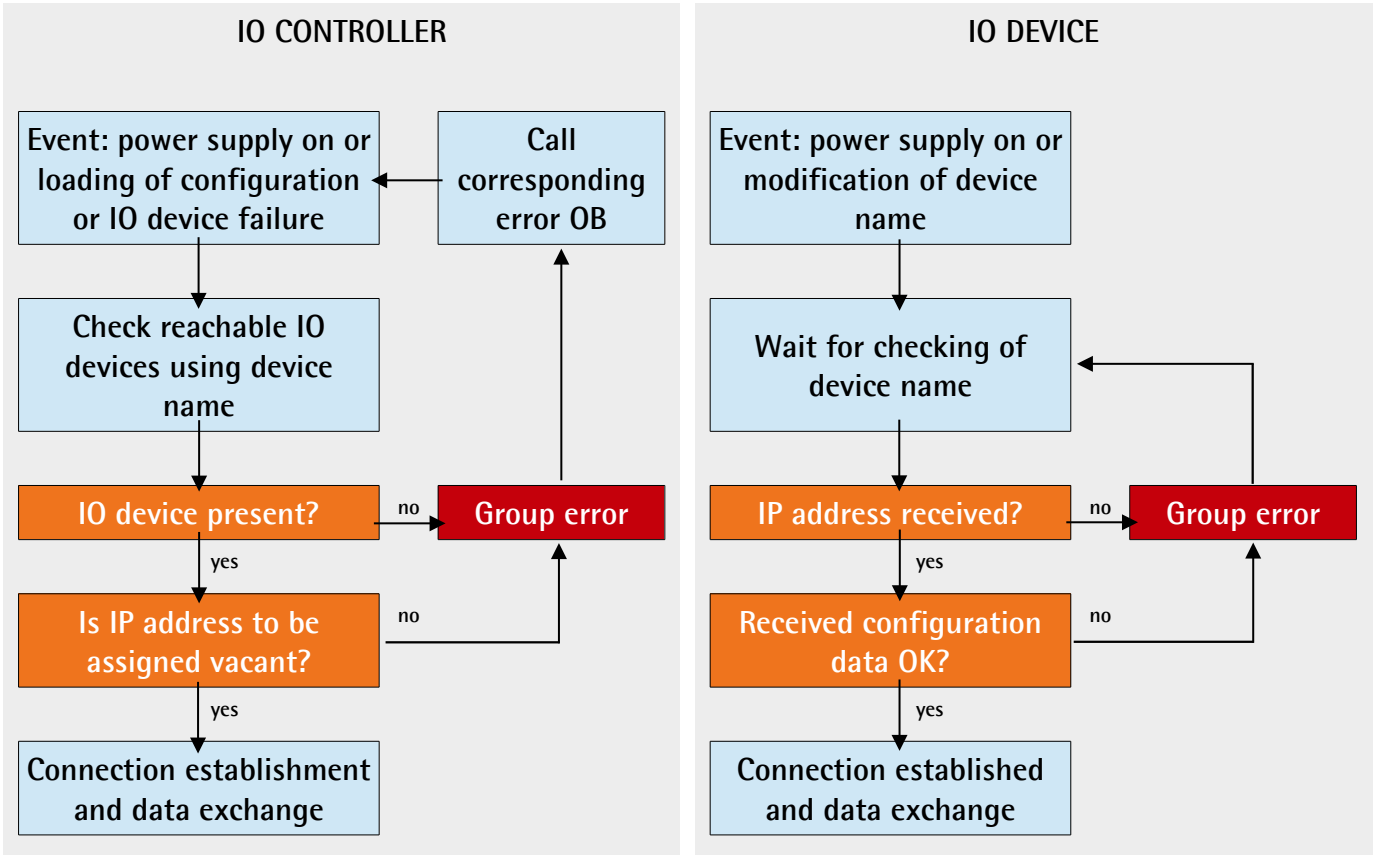
Press the **Device view** changeover switch in the **Hardware and network editor** to enter the **Device view** working area, then select the device you need to configure in the drop-down box on the top left of the graphic area. In the **Properties** inspector window, **General** tab, press the **Module parameters** menu option to see and set the converter's parameters if required.
The parameters listed in this page are sent in the Acyclic Data Exchange mode at switching on.
You can change the value of each parameter in the edit field. The new value will be transmitted to the Device in the Acyclic Data Exchange mode at switching on.
You can change the value of the module parameters also while the device is operational in the Cyclic Data Exchange mode via the Watch table. Please note that the value however will be overwritten at switching on by the value set in the Module parameters tabbed page.

For a comprehensive description of the parameters and how to set them properly refer to the specific explanation in the "9.4 Index 0xBF00 : user parameter data" section on page 86.

Steps for system start-up



Start-up response



6 Profinet interface

6.1 A brief introduction to Profinet

PROFINET IO is the open industrial network devised for automation applications and built on the Ethernet application layer (TCP/IP and IT standards). For PROFINET IO the layers 1 through 7a of the ISO/OSI (Open Systems Interconnection) reference model are exclusively based on internationally proven standards. The functionality of PROFINET is defined in layer 7b. PROFINET IO complies with IEEE802.3 Ethernet Standard and follows the standards IEC 61158 and IEC61784, so it is 100% Ethernet compatible.
Its technology development and standardization are entrusted to Profibus & Profinet International (PI), the international umbrella organization including members of more than 1400 companies (**www.profibus.com**).

PROFINET IO is expressly developed to connect controllers (named IO controllers, equivalent to Profibus DP Masters), peripheral devices (named IO devices, similar to Profibus DP Slaves) and programming devices / PCs (named IO supervisors) with Ethernet Real Time (RT) and Isochronous Real Time (IRT) communication all the way. Real Time channel is used for time-critical process data and allows to meet the real-time requirements of the automation engineering (cycle times < 500 µs, jitter < 1 µs); while IRT is suitable for sophisticated motion control and high performance applications in factory automation and permits cycle times lower than 250 µs with less than 1 µs jitter. The standard TCP/IP channel is used for parametrization, configuration and acyclic read/write operations.
A PROFINET IO system requires at least one IO Controller and one IO Device. The most frequent network topologies can be implemented and even mixed together including Star, Line, Tree and Ring structures by means of copper or fiber-optic cables. The number of devices (each one fitted with its own MAC address, IP address and device name) which can be connected in the PROFINET network is virtually unlimited. The transmission rate is 100Mbit/s with full duplex communication (Fast Ethernet).
PROFINET IO Devices are configured using a configuration tool which acts as the IO Supervisor. The IO Supervisor uses a GSD (General Station Description) file based on XML language, thus it is called GSDML file, see on page 50.

6.2 Profinet encoders and gateways from Lika Electronic

PROFINET encoders and gateways from Lika Electronic fully comply with the encoder profile specifications V4.1 version 3.162, the encoder profile is based on the PROFIdrive profile. For any information on the encoder profile please refer to the following document:

ENCODER PROFILE. Technical specifications for Profibus and Profinet related to PROFIdrive

edited by PI International.

Furthermore these encoders and gateways fulfil the requirements of the Application Classes 3 and 4, thus they are intended for clock-synchronous (isochronous) real-time applications with cyclic and synchronous data transmission. Anyway they can also be used in applications without clock synchronization. For detailed information on the application classes refer to the "6.3 Application Class definition" section on page 66.

PROFINET gateways support the standard telegram 81. Standard telegrams 82, 83 and 84 are not supported. Further information can be found in the "7.1 Telegrams" section on page 69.

The IO data is transferred to and from the Encoder Object (EO, see the "6.4 Encoder Object model" section on page 67) via the Cyclic Data Exchange Service. The EO comprises the following mandatory functionalities:

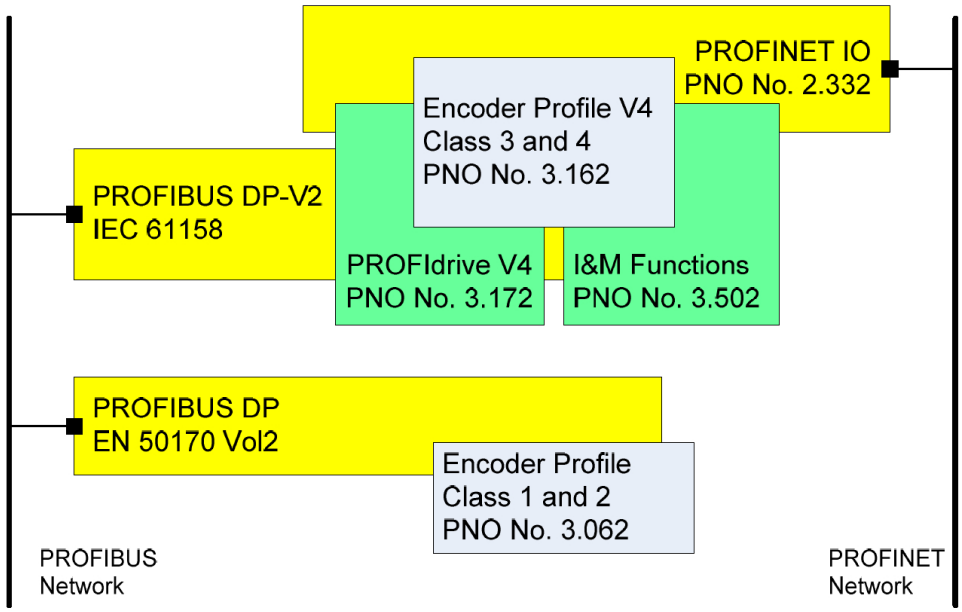
- parameters;
- measuring task (i.e. position value, ...);
- IO data (cyclical transmission of control and actual values);
- support for Alarm Mechanism.

Among the parameters available in the Profinet converters from Lika Electronic: code sequence, scaling function, preset (Class 4 functionalities), position readout, offset value, acyclic Error Data communication and diagnostic information.

PROFINET at a glance

Number of stations	Setting the IP-Address	Setting the baud rate	Transmission rate	Cable length	Cable
Virtually unlimited	Software / automatic via DCP	-	100 Mbit/s full duplex	Up to 100 m / 330 ft	M12 D-coded Profinet connectors

6.2.1 Overview of the encoder profiles



6.3 Application Class definition

The encoder profile defines two application classes: **Class 3** and **Class 4**. A number of mandatory functions are specified for each application class, in addition all optional functions must be recognized by the encoder and handled so that the controller is able to determine whether an optional function is supported.

NOTE
There is no relation between the Encoder application classes and the application classes defined in the PROFIdrive profile.

6.3.1 Application Class 3

Encoder with base mode parameter access and limited parametrization of the encoder functionality. Isochronous mode is not supported.

6.3.2 Application Class 4

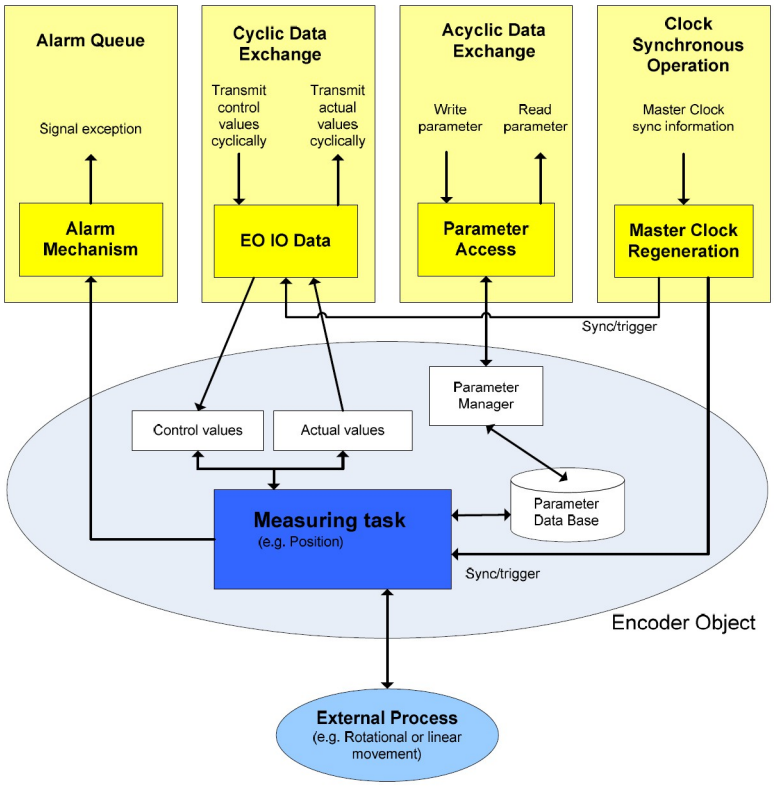
Encoder with scaling, preset, isochronous mode and base mode parameter access. A Class 4 configured encoder fully supports all functionalities according to the encoder profile V4.1.

Lika Electronic encoders fulfil the requirements of CLASS 4

6.4 Encoder Object model

The Figure shows the general Encoder Object (EO) architecture. Central element of the EO is the Measuring Task where the measurements are made and the results are calculated. The properties of the EO is represented and controlled by parameters. The parameters are administered in the Parameter Data Base. To access EO parameters, Acyclic Data Exchange service is used. For periodic transportation of control values to the EO and actual values from the EO, the Cyclic Data Exchange service is used. Exception situations out of the Measuring Task and the General State Machine may be signalled by the Alarm Mechanism to the controlling device.
The EO shall comprise as minimum mandatory functionality:

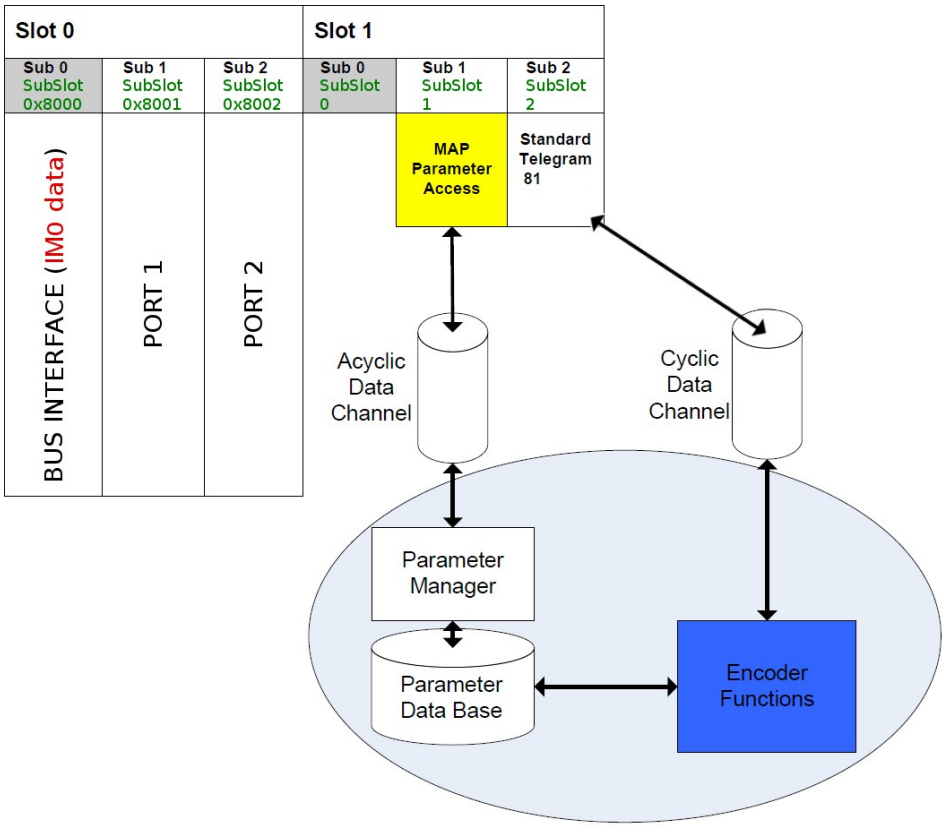
- Parameters;
- Measuring Task;
- IO Data (control value, actual value);
- Support for Alarm Mechanism;
- Optional functionality;
- Clock Synchronous operation.



6.5 Encoder object architecture

The Figure shows the general architecture and the mapping of the Encoder Object (EO) architectural elements to Communication Objects of the Peripheral Device for PROFINET IO. General with PROFINET IO the EO is mapped exactly to one Module/Slot. Slot 0 is exclusively reserved for Device representative purpose and therefore shall not used for any Encoder module. Valid Slot numbers for Encoder Objects are from 1 to 0x7FFF. Every EO contains at least the mandatory Module Access Point (MAP) which is mapped to a dedicated EO representative Submodule. This MAP Submodule contains at least the mandatory Parameter Access Point (PAP) which is mapped to a dedicated Record Data Object. Via the EO representative Submodule (MAP) and the specified Record Data Object the access to the EO parameter manager is possible. The EO parameter manager has access to the EO local Parameter Data Base. In addition to the mandatory MAP submodule, the EO may contain additional submodules which may be used to:

- represent communication end points for IO Data (cyclic data channel) and also to structure the IO Data in data blocks (telegrams, signals).
- represent physical or logical Subobjects of the EO.



7 PROFINET IO data description

7.1 Telegrams

A telegram is a rigidly defined bit stream carrying data. In each telegram the data length and the type of data which is sent to and from the IO controller is specified. PROFINET interface devices communicate and stay in sync by sending each other telegrams. IF55 gateway supports one only type of telegram: Standard Telegram 81. It is described hereafter. Standard telegrams 82, 83 and 84 are not supported. Standard signals are fully described in the "Cyclic Data Exchange – Std signals" section on page 70.

7.1.1 Standard Telegram 81

The Standard Telegram 81 uses 4 bytes to output data from the IO controller to the encoder and 12 bytes to input data from the encoder to the IO controller.

Output data CONTROLLER => DEVICE

	2 bytes	2 bytes
IO Data	1	2
Set point	**STW2_ENC**	**G1_STW**

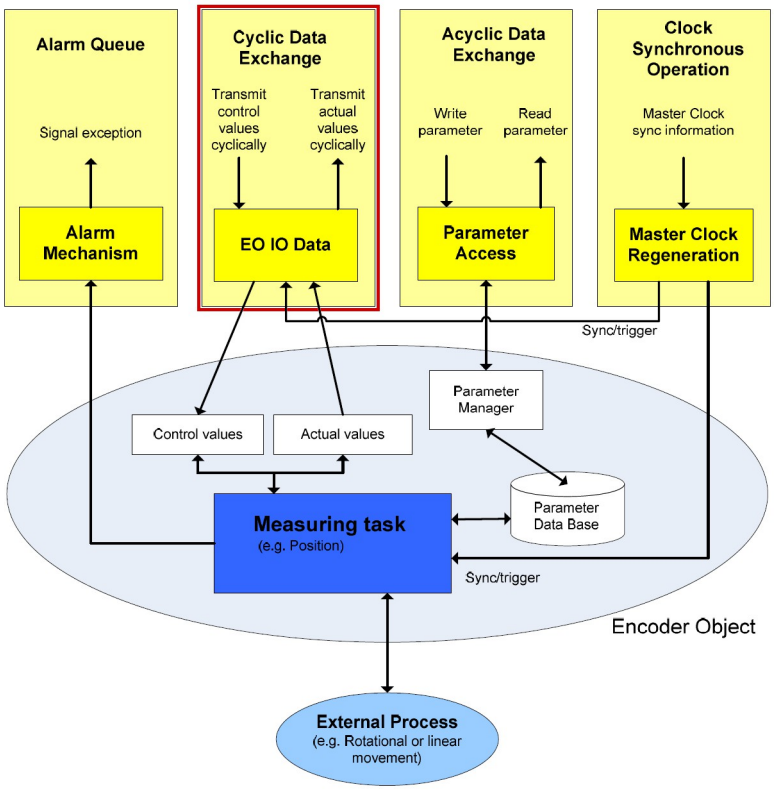
Input data DEVICE => CONTROLLER

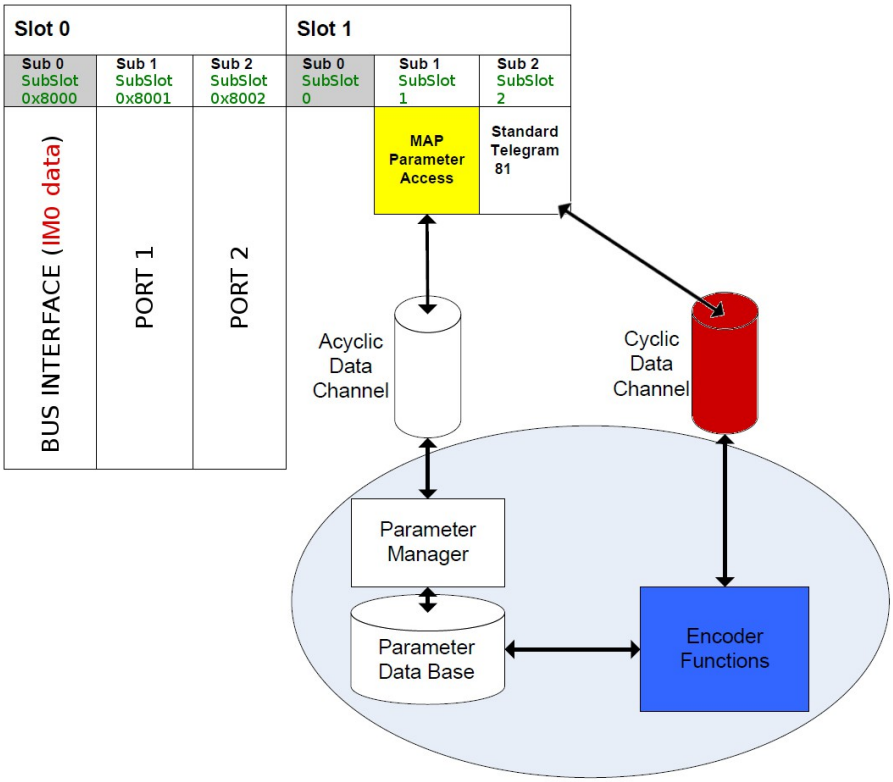
	2 bytes	2 bytes	4 bytes		4 bytes	
IO Data	1	2	3	4	5	6
Actual value	**ZSW2_ENC**	**G1_ZSW**	**G1_XIST1**		**G1_XIST2**	

8 Cyclic Data Exchange – Std signals

IO data is transferred via the Cyclic Data Exchange. A series of standard signals are defined to configure the IO data. In the following table the standard signals are summarily described.

Significance	Abbreviation	Length (bits)	Data type	Page
Sensor 1 position current value 1	**G1_XIST1**	32	Unsigned	72
Sensor 1 position current value 2	**G1_XIST2**	32	Unsigned	73
Encoder Control word 2	**STW2_ENC**	16	Unsigned	73
Encoder Status word 2	**ZSW2_ENC**	16	Unsigned	74
Sensor 1 Control word	**G1_STW**	16	Unsigned	75
Sensor 1 Status word	**G1_ZSW**	16	Unsigned	78




Slot 0
Slot 1
Sub 0 SubSlot 0x8000
Sub 1 SubSlot 0x8001
Sub 2 SubSlot 0x8002
Sub 0 SubSlot 0
Sub 1 SubSlot 1
Sub 2 SubSlot 2
BUS INTERFACE (IM0 data)
PORT 1
PORT 2
MAP Parameter Access
Standard Telegram 81
Acyclic Data Channel
Cyclic Data Channel
Parameter Manager
Parameter Data Base
Encoder Functions

8.1 List of the available standard signals

G1_XIST1

[Unsigned, 32 bits]
It is defined as Sensor 1 current position value 1. This signal is the current (real) absolute position of the encoder expressed in binary notation.

Format definition:

- all values are represented in binary notation;
- the recommended default shift factor is zero (right aligned value) for both **G1_XIST1** and **G1_XIST2**;
- the settings in the encoder parameter data affect the position value in both **G1_XIST1** and **G1_XIST2**.

EXAMPLE
Here follows a format example.
25-bit absolute multiturn encoder, 13-bit singleturn resolution (8,192 counts per revolution), 12-bit multiturn resolution (4,096 revolutions)

M	= Multiturn value, number of revolutions
S	= Singleturn value, number of counts per revolution

MSB LSB

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
							M	M	M	M	M	M	M	M	M	M	M	M	S	S	S	S	S	S	S	S	S	S	S	S	S

Absolute position value in **G1_XIST1**

MSB LSB

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
							M	M	M	M	M	M	M	M	M	M	M	M	S	S	S	S	S	S	S	S	S	S	S	S	S

Absolute position value in **G1_XIST2**

G1_XIST2

[Unsigned, 32 bits]
It is defined as Sensor 1 current position value 2. By default this signal is the current (real) absolute position of the encoder expressed in binary notation yet it has a different meaning if an error is active.

If no error is active:
this signal informs about the current position value of the encoder, provided that the bit **Request absolute value cyclically** (bit 13 of control word **G1_STW**) is set to 1; otherwise this value is 0.

If an error is active:
this signal informs about the active error. For the complete list of the error codes refer to the "10.3 Error codes in G1_XIST2" section on page 116.

Format definition:

- all values are represented in binary notation;
- the recommended default shift factor is zero (right aligned value) for both **G1_XIST1** and **G1_XIST2**;
- the settings in the encoder parameter data affect the position value in both **G1_XIST1** and **G1_XIST2**;
- **G1_XIST2** displays the error telegram instead of the position value if an error occurs.

For the format example see **G1_XIST1** above.

STW2_ENC

[Unsigned, 16 bits]
It is defined as Encoder control word 2. Control word **STW2_ENC** includes the **Control by PLC** mechanisms from PROFIdrive STW1 and the **Controller Sign-Of-Life** mechanism from PROFIdrive STW2.

Bit	Meaning
0 ... 6	Reserved
7	Not used
8 & 9	Reserved
10	**Control by PLC**
11	Reserved
12 ... 15	**Controller Sign-Of-Life**

Control by PLC
Bit 10
If the **Compatibility Mode** is enabled (see on page 90), then bit 10 **Control by PLC** is ignored. In this case control word **G1_STW** and setpoint are always checked.

If the **Compatibility Mode** is disabled (see on page 90), then bit 10 **Control by PLC** is checked. So control word **G1_STW** and setpoint are checked only if the bit **Control by PLC** is set.

Bit	Value	Significance	Comment
10	1	Control from PLC	Control via interface, EO IO Data is processed.
	0	No control from PLC	EO IO Data not valid, except Sign-Of-Life.

Controller Sign-Of-Life
Bits 12 ... 15

For more information on the control word **STW2_ENC** please refer to the PROFIdrive Technical Specification document.

ZSW2_ENC
[Unsigned, 16 bits]
It is defined as Encoder status word 2. The encoder status word 2 **ZSW2_ENC** includes the Control by PLC mechanism from PROFIdrive ZSW1 and the Slave Sign-Of-Life mechanism from PROFIdrive ZSW2.

Bit	Meaning
0 ... 2	Reserved
3	Not used
4 ... 8	Reserved
9	**Control requested**
10 & 11	Reserved
12 ... 15	**Encoder Sign-Of-Life**

Control requested
Bit 9

Bit	Value	Significance	Comment
9	1	Control Requested	The automation system is requested to assume control.
	0	No Control requested	Control by the automation system is not possible, only possible at the device or by another interface.

Encoder Sign-Of-Life
Bits 12 ... 15

For more information on the status word 2 **ZSW2_ENC** please refer to the PROFIdrive Technical Specification document.

G1_STW

[Unsigned, 16 bits]
It is defined as Sensor 1 control word. This control word controls the functionality of major encoder functions.

Bit	Meaning
0 ... 7	Not used
8 ... 10	Reserved
11	**Home position mode**
12	**Request set/shift of home position**
13	**Request absolute value cyclically**
14	**Activate parking sensor**
15	**Acknowledging a sensor error**

NOTE
If the **Activate parking sensor** is activated (bit 14 = 1) the encoder is still on bus with the Slave Sign-Of-Life active and the encoder error and diagnostics switched off.

Home position mode
Bit 11

Request set/shift of home position
Bit 12

The preset function is controlled by bits 11 and 12 in this Sensor 1 control word **G1_STW** and acknowledged by the bit 12 **Set/shift of home position executed** in the sensor status word **G1_ZSW**. The preset value is 0 by default and may be set by an acyclic data exchange parameter defined in the parameters section (see **P65000 – Preset value** on page 83). The preset function has an absolute and a relative operating mode selectable by mean of the bit 11 **Home position mode** in this Sensor 1 control word **G1_STW** (0 = absolute; 1 = relative). Bit 11 and bit 12 in the Sensor 1 control word **G1_STW** control the preset function as described in the table below.

Bit 12	Bit 11	Action
0	X	**Normal operating mode.** The encoder will make no change in the output value.
1	0	**Preset mode absolute** The encoder reads the current position value and calculates an internal offset value from the preset value **P65000 – Preset value** and the read position value. The position value is then shifted with the calculated offset value to get the current position value equal to the preset value. The encoder acknowledges the preset by setting the bit 12 **Set/shift of home position executed** in the sensor status word **G1_ZSW**. Now bit 12 **Request set/shift of home position** in the sensor 1 control word **G1_STW** can be set to zero by the Master. The encoder will end the preset cycle by clearing the bit 12 **Set/shift of home position executed** in the sensor status word **G1_ZSW**. The new internal offset value can be read with an acyclic data exchange parameter (if implemented) and is securely stored in case of voltage breakdown and uploaded again at each power on.
1	1	**Preset mode relative (offset)** The encoder uses the preset value **P65000 – Preset value** as a relative offset value. In this mode the current position value is shifted by the value deriving from the preset value. **EXAMPLE** A preset value "1000" is intended to shift the current position value by 1000 steps in the positive counting direction. So a "real" position value of "5000" will have the value "6000" after the relative shifting sequence. The encoder will set bit 12 **Set/shift of home position executed** in the sensor status

		word **G1_ZSW** to acknowledge the execution of the shifting. Bit 12 **Request set/shift of home position** in the sensor control word **G1_STW** can now be set to zero by the Master. The encoder will end the preset cycle by clearing the bit 12 **Set/shift of home position executed** in the sensor status word **G1_ZSW**. The internal offset value will be shifted according to the transferred preset value. The new offset value is securely stored in case of voltage breakdown and uploaded again at each power on.

The Preset command automatically saves the calculated internal offset values.

NOTE
Refer also to the index **P65000 – Preset value** on page 83; to G1_XIST1 Preset control on page 89; and to the "14.2 Preset diagram" section on page 155. See also the "15.2 Setting the preset value" section on page 171.

EXAMPLE
An example of setting the Preset value is provided on page 146. Refer also to the "5.1.4 Setting and executing the preset" section on page 38.

Request absolute value cyclically
Bit 13

Bit	Significance	Comment
13	=1 : **Request absolute value cyclically**	Request of additional cyclic transmission of the current absolute position in **G1_XIST2**.

Activate parking sensor
Bit 14

Bit	Significance	Comment
14	=1 : **Activate parking sensor**	Request to stop monitoring the measuring system and the current value measurements in the drive. This makes it possible to disconnect the encoder from the line without having

		to change the drive configuration or causing a fault. In this case all current errors of the encoder are cleared. The parking of the encoder while the drive is running is not allowed and will result in a sensor interface error (error code 0x03 in **G1_XIST2**).

See also "14.3 Parking sensor diagram" section on page 156.

Acknowledging a sensor error
Bit 15

Bit	Significance	Comment
15	=1 : **Acknowledging a sensor error**	Request to acknowledge a sensor error (bit 15 **Sensor error** of **G1_ZSW**).

G1_ZSW

[Unsigned, 16 bits]
It is defined as Sensor 1 status word. This status word defines the states, acknowledgements and error messages of the encoder and its main functions.

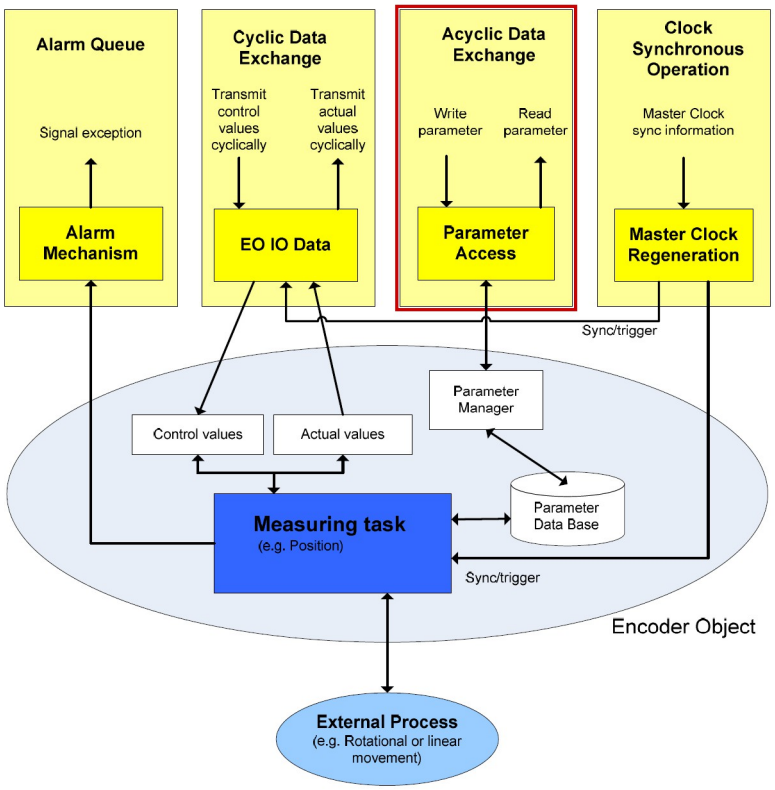
Bit	Meaning
0 ... 9	Not used
10	Reserved
11	**Requirements of error acknowledge detected**
12	**Set/shift of home position executed**
13	**Transmit absolute value cyclically**
14	**Parking sensor active**
15	**Sensor error**

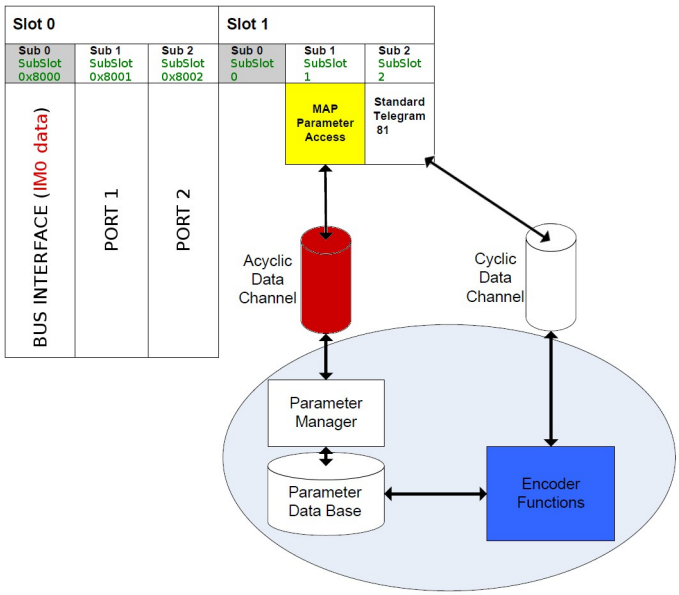
NOTE
If bit13 **Transmit absolute value cyclically** or bit15 **Sensor error** are not set, there is no valid value or error code transferred in **G1_XIST2**.

NOTE
Bit 13 **Transmit absolute value cyclically** and bit 15 **Sensor error** cannot be set at the same time as they are used to indicate either a valid position value transmission (bit 13) or the error code transmission (bit 15) in **G1_XIST2**.

9 Acyclic Data Exchange

In addition to the Cyclic Data Exchange (see the "Cyclic Data Exchange – Std signals" section on page 70), the Acyclic Data Exchange gives the possibility to read and write parameters over the non real time channel.





9.1 Index 0xAFF0: Identification & Maintenance (I&M) functions

Profinet encoders/gateways from Lika Electronic only implement I&M 0 Module (IM0).

IM0 is accessible with record 0xAFF0 and provides general information on the device such as vendor ID, order ID, serial number, etc.

Description	Number of bytes
BLOCKHEADER	6
MANUFACTURER ID (VENDOR ID)	6
ORDER ID	20
SERIAL NUMBER	16
HARDWARE REVISION	2
SOFTWARE REVISION	4
REVISION COUNTER	2
PROFILE ID (API)	2
PROFILE SPECIFIC TYPE	2
IM VERSION	2
IM SUPPORTED	2

9.2 Index 0xB02E : supported PROFIdrive specific parameters

P922 – Telegram Selection

[Unsigned16, RO]

It indicates the type of telegram which is currently in use. Possible value: 81 (telegrams 82, 83 and 84 not implemented).

P964 – Profidrive Parameter : Device identification

[Array[0 ... 5], unsigned16, RO]

Index	Sub	Meaning	Value	Access
964	0	Manufacturer ID (Vendor ID assigned by PI)	0x239	RO
964	1	DU Drive unit type (Vendor specific)	1	RO
964	2	Software version	xxxx	RO
964	3	Software year	yyyy	RO
964	4	Software day and month	dd.mm	RO
964	5	Number of Drive Object (DO)	1	RO

P965 – Encoder profile number

[Octet string 2, RO]

Index	Sub	Meaning	Value	Access
965	0	Encoder profile number	0x3D	RO
965	1	Encoder profile version, set by customer	31 or 41	RO

P971 – Transfer to non volatile memory

[Unsigned16, RW]

It is used to save the current local parameters on a non volatile memory. Write "1" to save the parameters. The encoder confirms save by writing "0" into this parameter.

Index	Sub	Meaning	Value	Access
971	0	Save on non volatile memory	variable	RW

P975 – Encoder object identification

[Array[0 ... 7], unsigned16, RO]

Index	Sub	Meaning	Value	Access
975	0	Manufacturer ID (Vendor ID assigned by PI)	0x239	RO
975	1	DO type (Vendor specific)	0x01	RO
975	2	Software version	xx.xx	RO
975	3	Software year	yyyy	RO
975	4	Software day and month	dd.mm	RO
975	5	Profidrive DO type classification	0x05 encoder interface	RO
975	6	Profidrive DO subclassification 1	0x8000 (encoder Application Class 4 supported)	RO
975	7	Drive object ID (DO ID)	0x01	RO

P979 – Sensor format

[Array[0 ... 5], unsigned16, RO]

Index	Sub	Meaning	Value	Access
979	0	Header	0x00005011	RO
979	1	Sensor type	0x80000000	RO
979	2	Sensor Resolution	variable	RO
979	3	Shift factor for **G1_XIST1**	0	RO
979	4	Shift factor for **G1_XIST2**	0	RO
979	5	Determinable resolutions	variable	RO

P980 – Number list of defined parameter

[Array[0 ... 8], unsigned16, RO]

Index	Sub	Meaning	Value	Access
980	0	**P922 – Telegram Selection**	922	RO
980	1	**P964 – Profidrive Parameter : Device identification**	964	RO
980	2	**P965 – Encoder profile number**	965	RO
980	3	**P971 – Transfer to non volatile memory**	971	RO
980	4	**P975 – Encoder object identification**	975	RO
980	5	**P979 – Sensor format**	979	RO
980	6	**P61001 – IP of station**	61001	RO
980	7	**P65000 – Preset value**	65000	RO
980	8	**P65001 – Operating status**	65001	RO

P61001 – IP of station

[Unsigned32, RO]

Index	Sub	Meaning	Value	Access
61001	0	IP address assigned to the encoder	variable	RO

9.3 Index 0xB02E : supported encoder specific parameters

P65000 – Preset value

[Unsigned32, RW]

Preset mode absolute

Preset function is meant to assign a desired value to a known physical position of the system. The chosen physical position will get the value set next to this index and all the previous and following mechanical positions will get a value according to it.

Preset mode relative (offset)

The encoder uses the preset value **P65000 – Preset value** as a relative offset value. In this mode the current position value is shifted by the value deriving from this preset value.

Preset value can be saved on the internal memory using the parameter **P971 – Transfer to non volatile memory**.

See also **Home position mode** and **Request set/shift of home position** in **G1_STW** on page 75; and G1_XIST1 Preset control on page 89.

Index	Sub	Meaning	Value	Access
65000	0	Preset value	variable	RW

EXAMPLE

An example of setting the Preset value is provided on page 146.

NOTE

Please consider that if the Bypass parameter (see on page 92) is set to "1" = enabled, the preset function -even if set and activated- is ignored. If the user sets a preset while the "Bypass mode" is enabled, the operation is not carried out.

P65001 – Operating status

[Array[0 ... 11], unsigned32, RO]

This parameter has a read only structure where information on the encoder operating status can be found. It is a complement to the PROFIdrive parameter **979** described in the PROFIdrive profile.

Index	Sub	Meaning	Value	Access
65001	0	**Header**	0x000C0101	RO
65001	1	**Operating status**	See "Operating status table values"	RO
65001	2	**Faults**	See "Faults table"	RO
65001	3	**Supported faults**	0x0030	RO
65001	4	**Warnings** (*)	0x0	RO
65001	5	**Supported warnings**	0x0	RO
65001	6	**Encoder profile version** (**)	0x401	RO
65001	7	**Operating time**	0xFFFFFFFF	RO
65001	8	**Offset value** (related to **G1_XIST1**)	variable	RO
65001	9	**MeasuringUnitsPerRev** = Physical pulse resolution [nm] ***	variable	RO
65001	9	**MeasuringUnitsPerRev** = Physical singleturn resolution [bit] ****	variable	RO
65001	10	**Total Physical measuring range**	variable	RO

(*) Warnings are not supported in this encoder.

(**) The encoder profile version is the version of the encoder profile document implemented in the encoder. This parameter is not affected by the Compatibility Mode settings.

(***) IF55 LIN gateway only (for linear encoders)

(****) IF55 ROT gateway only (for rotary encoders)

Operating status table values

Bit	Meaning
0	Code sequence
1	Class 4 functionality
2	G1_XIST1 Preset control
3	Scaling function control
4	Alarm channel control

5	Compatibility Mode
6 & 7	Reserved to the encoder manufacturer
8 ... 31	Reserved for future use

Faults table

Bit	Meaning
0 ... 3	Not used
4	Commissioning diagnostic
5	Memory error
6 ... 31	Not used

Supported faults table

Bit	Meaning
0 ... 3	Not used
4	Commissioning diagnostics supported
5	Memory error supported
6 ... 31	Not used

Offset value is calculated in the preset function and is intended to shift the position value. The offset value is saved on the internal memory. This parameter is a read-only parameter.

9.4 Index 0xBF00 : user parameter data

The 31-byte user parameter data listed in the tables below is sent to the encoder in the start-up phase using the data record 0xBF00.
In the following table (see "9.4.1 List of the common parameters" section) the parameters that are common to both rotary and linear encoders are listed.

9.4.1 List of the common parameters

For a comprehensive description of the the listed parameters refer to the "9.4.4 Description of the common parameters" section on page 88.

Parameter	Data Type	Default	Comment	User Data Octet Number	
Code sequence	Bit	0 (CW)		Byte 0	bit 0
Class 4 functionality	Bit	1 (enabled)			bit 1
G1_XIST1 Preset control	Bit	0 (enabled)			bit 2
Scaling function control	Bit	0 (disabled)			bit 3
Alarm channel control	Bit	0 (disabled)	Only supported in Compatibility Mode		bit 4
Compatibility Mode	Bit	1 (disabled) (profile V4.1)			bit 5
Reserved		0	Set to 0		bits 6-7
Maximum tolerated failures of Master Sign-Of-Life	Unsigned8	1	Only supported in Compatibility Mode	Byte 1	
Code Format	Bit	0 (Binary)		Byte 2	bit 0
Reserved		0	Set to 0		bits 1-7
Alignment	Bit	0 (LSB Right Alignment)		Byte 3	bit 0
Bypass	Bit	0 (disabled)			bit 1
Reserved		0	Set to 0		bits 2-7
Number of clocks	Unsigned8	32		Byte 4	
Parameters specific to rotary and linear encoders. Refer to "9.4.2 List of the parameters specific to linear encoders (DAP 1)" and "9.4.3 List of the parameters specific to rotary encoders (DAP 2)" sections.				Bytes 5 ... 30	

9.4.2 List of the parameters specific to linear encoders (DAP 1)

For a comprehensive description of the the listed parameters refer to the "9.4.5 Description of the parameters specific to the linear encoders (DAP1)" section on page 95.

Parameter	Data Type	Default	Comment	User Data Octet Number
Physical pulse resolution [nm]	Unsigned32	1,000,000		Bytes 5 ... 8
Programmable pulse resolution [nm]	Unsigned32	1,000,000		Byte 9 ... 12
Physical Total resolution [bit]	Unsigned32	30		Bytes 13 ... 16
Programmable Total resolution [pulse]	Unsigned32	1,073,741,824		Bytes 17 ... 20
Reserved		0x00	Set to 0	Bytes 21 ... 30

9.4.3 List of the parameters specific to rotary encoders (DAP 2)

For a comprehensive description of the the listed parameters refer to the "9.4.6 Description of the parameters specific to the rotary encoders (DAP2)" section on page 105.

Parameter	Data Type	Default	Comment	User Data Octet Number
Physical singleturn resolution [bit]	Unsigned8	16		Byte 5
Physical multiturn resolution [bit]	Unsigned8	14		Byte 6
Programmable pulse/rev	Unsigned32	65,536		Bytes 7 ... 10
Programmable total measuring range (pulse)	Unsigned32	1,073,741,824		Bytes 11 ... 14
Reserved		0x00	Set to 0	Bytes 15 ... 30

NOTE
Default values are highlighted in **bold** in the following tables.

9.4.4 Description of the common parameters

Code sequence

In rotary encoders **Code sequence** sets whether the absolute position value output by the encoder (count up information) increases when the encoder shaft rotates clockwise (0 = CW) or counter-clockwise (1 = CCW). CW and CCW rotations are viewed from shaft end.
In linear encoders **Code sequence** sets whether the absolute position value output by the encoder (count up information) increases when the encoder moves in the standard direction (it is indicated in the encoder's manual) or when the encoder moves in reverse of the standard direction.
This parameter is processed only if **Class 4 functionality** is enabled.
Default = 0 (min. = 0, max. = 1)

WARNING
Changing this value causes also the position calculated by the controller to be necessarily affected. Therefore it is mandatory to execute a new preset after setting this parameter.

NOTE
Please consider that if the **Bypass** parameter (see on page 92) is set to "1" = enabled, the counting direction function -if set differently from default- is ignored.

Class 4 functionality

For any information on the implemented Application Classes refer to the "6.3 Application Class definition" section on page 66.
If it is enabled, **Code sequence**, **G1_XIST1 Preset control** and **Scaling function control** affect the position value in **G1_XIST1** and **G1_XIST2**. However the preset will not affect the position value in **G1_XIST1** if the parameter **G1_XIST1 Preset control** is disabled; it will always affect **G1_XIST2** instead.

Attribute	Meaning	Value
Disabled	**Code sequence**, **G1_XIST1 Preset control** and **Scaling function control** disabled	0
Enabled	**Code sequence**, **G1_XIST1 Preset control** and **Scaling function control** enabled	**1**

G1_XIST1 Preset control

This parameter is available only if **Class 4 functionality** is enabled.
This parameter controls the effect of a preset on the **G1_XIST1** current value. When it is enabled, the Preset will affect the position value in **G1_XIST1**.

Attribute	Meaning	Value
Enabled	**G1_XIST1** is affected by a Preset command	**0**
Disabled	Preset does not affect **G1_XIST1**	1

WARNING
G1_XIST1 Preset control is disabled by setting the value 1.

NOTE
There is no functionality of this parameter if the **Class 4 functionality** parameter is disabled.

EXAMPLE
An example of setting the Preset value is provided on page 146.

Scaling function control

This parameter enables / disables the Scaling function.
When this parameter is disabled (0), the device uses the **hardware** resolutions (refer to **Physical singleturn resolution [bit]** and **Physical multiturn resolution [bit]** parameters for rotary encoders; refer to **Physical pulse resolution [nm]** and **Physical Total resolution [bit]** parameters for linear encoders).
When this parameter is enabled (1), the device uses the custom resolutions set next to **Programmable pulse/rev** and **Programmable total measuring range (pulse)** parameters (rotary encoders); next to **Programmable pulse resolution [nm]** and **Programmable Total resolution [pulse]** parameters (linear encoders).

Attribute	Meaning	Value
Disabled	Scaling function disabled	**0**
Enabled	Scaling function enabled	1

NOTE
There is no functionality of this parameter if the **Class 4 functionality** parameter is disabled.

NOTE
Please consider that if the **Bypass** parameter (see on page 92) is set to "1" = enabled, the scaling function -if set differently from default- is ignored.

Alarm channel control

This parameter enables / disables the encoder specific Alarm channel transferred as Channel Related Diagnosis. This functionality is used to limit the amount of data sent in isochronous mode.
If the value is zero (0 = default value), only the communication related alarms are sent via the alarm channel. If the value is one (1), also the encoder specific faults and warnings are sent via the alarm channel.
For further information refer also to the "10.2 Error messages via the Alarm Channel" section on page 115.

Attribute	Meaning	Value
Disable	No profile specific diagnosis	**0**
Enable	Profile specific diagnosis	1

NOTE
This parameter is only supported in compatibility mode (see **Compatibility Mode** parameter hereafter).

Compatibility Mode

This parameter defines whether the encoder has to run in a mode compatible with Version 3.1 of the Encoder Profile. See the table below for an overview of the functions affected when the compatibility mode is enabled.

Attribute	Meaning	Value
Enabled	Compatibility with Encoder Profile V3.1	0
Disabled	No backward compatibility	**1**

Function	Compatibility mode Enabled (=0)	Compatibility mode Disabled (=1)
Control by PLC (**STW2_ENC**)	Ignored. The control word **G1_STW** and setpoint values are always valid. **Control requested** (**ZSW2_ENC**) is not supported and is set to 0	Supported
User parameter **Maximum tolerated failures of Master Sign-Of-Life**	Supported	Not supported. One Sign-Of-Life failure tolerated. **P925** is optional to control the life sign monitoring
User parameter **Alarm channel control**	Supported	Not supported. The application alarm channel is active and controlled by a PROFIdrive parameter
P965 – Encoder profile number	31 (V3.1)	41 (V4.1)

Maximum tolerated failures of Master Sign-Of-Life

With this parameter the number of allowed failures of the Master's sign of life is defined. The default value is one (1).
Default = 1 (min. = 1, max. = 255)

NOTE
This parameter is only supported in compatibility mode (see **Compatibility Mode** parameter in the previous page).

Code Format

It sets the output code used by the SSI encoder to output the absolute position information. The output code can be Binary (bit 0 **Code Format** = 0) or Gray (bit 0 **Code Format** = 1). For any information on the output code please refer to the "User's manual" of the connected encoder.
Default = 0 = Binary (min. = 0 = Binary, max. = 1 = Gray)

EXAMPLE
We need to connect the following rotary encoder: **MM36 12/8192 BB**.
MM36 ... BB encoder uses the Binary code to output the absolute position information. Thus you have to set the value 0 = Binary in this bit. For further information refer to the "User's manual" of the connected encoder.

EXAMPLE
We need to connect the following linear encoder: **SMA5-GA-50**.
SMA5 encoder uses the Gray code to output the absolute position information. Thus you have to set the value 1 = Gray in this bit. For further information refer to the "User's manual".

Alignment

It sets the SSI protocol used by the SSI encoder to arrange the absolute position information. The SSI protocol can be the "LSB Right Aligned" protocol (bit 0 **Alignment** = 0) or the "MSB Left Aligned" protocol (bit 0 **Alignment** = 1). For any information on the SSI protocol please refer to the "User's manual" of the connected encoder.
Default = 0 (min. = 0, max. = 1)

EXAMPLE
We need to connect the following rotary encoder: **MM36 12/8192 BB**.
MM36 encoder uses the 25-bit "LSB Right Aligned" protocol to arrange the absolute position information. Thus you have to set the value 0 in this bit. For further information refer to the "User's manual" of the connected encoder.

EXAMPLE
We need to connect the following linear encoder: **SMAX-BG-100**.
"BG" in the order code means that "MSB Left Aligned" protocol and Binary code are used to arrange the absolute position information. Thus you have to set the value 1 in this bit. For further information refer to the "User's manual".

Bypass

If the bit 1 **Bypass** = 0 = disabled, the "Bypass mode" is disabled, that is: the position value (refer to the **G1_XIST1** parameter on page 72) read by the encoder can be processed according to needs, so the user can scale the value, set a preset and change the counting direction.
If the bit 1 **Bypass** = 1 = enabled, the "Bypass mode" is enabled, that is: the information from the encoder is transmitted "as it is" and not processed in any way. The preset, scaling and counting direction functions -even if set and

enabled- are ignored. If, for example, the user sets a preset while the "Bypass mode" is enabled, the value is accepted, but not activated. As soon as the "Bypass mode" is disabled, the preset, scaling and counting direction functions -if set and enabled- become active and the position value will be accordingly.
Default = 0 (min. = 0, max. = 1)

Number of clocks

It sets the number of SSI clocks required by the SSI encoder to send the complete data word. The number of clocks depends on the resolution of the encoder / the max. number of information and the type of SSI protocol. The value has to be comprised between 1 and 32. For any information on the SSI clocks required please refer to the "User's manual" of the connected encoder.
Default = 32 (min. = 1, max. = 32)

WARNING

If **Alignment** = 1 = "MSB Left Aligned" protocol, the **Number of clocks** must be equal to the sum of the bits of the single- and multiturn physical resolutions (**Physical singleturn resolution [bit]** + **Physical multiturn resolution [bit]**) for rotary encoders; equal to number of bits of the total physical resolution (**Physical Total resolution [bit]**) for linear encoders.

EXAMPLE

We need to connect the following rotary encoder: **AS58 13/BB**.
AS58 uses the 13-bit "LSB Right Aligned" protocol to arrange the absolute position information as its overall resolution is ≤ 13 bits (13 bits). It always requires 13 clocks (the length of the word is always 13 bits, regardless of the max. number of information to provide). Thus you have to set 13 in this parameter. For further information refer to the "User's manual" of the connected encoder.

EXAMPLE

We need to connect the following rotary encoder: **MM36 12/8192 BB**.
MM36 uses the 25-bit "LSB Right Aligned" protocol to arrange the absolute position information as its overall resolution is ≤ 25 bits (12 + 13 bits). It always requires 25 clocks (the length of the word is always 25 bits, regardless of the max. number of information to provide). Thus you have to set 25 in this parameter. For further information refer to the "User's manual" of the connected encoder.

EXAMPLE

We need to connect the following rotary encoder: **HM58 16/16384 BA**.
HM58 uses the 32-bit "LSB Right Aligned" protocol to arrange the absolute position information as its overall resolution is ≤ 32 bits (16 + 14 bits). It always requires 32 clocks (the length of the word is always 32 bits, regardless of the max. number of information to provide). Thus you have to set 32 in this parameter. For further information refer to the "User's manual" of the connected encoder.

EXAMPLE

We need to connect the following linear encoder: **SMA5-GA-50**.
SMA5 encoder always requires 25 clocks (the length of the word is always 25 bits, regardless of the max. number of information to provide). Thus you have to set 25 in this parameter. For further information refer to the "User's manual".

EXAMPLE

We need to connect the following linear encoder: **SMAX-BG-100**.
The number of clocks depends on the max. number of information (see the example in the following parameter). Let's say the max. number of information is 6,000, thus it requires 13 clocks. You have to set 13 in this parameter. For further information refer to the "User's manual".

9.4.5 Description of the parameters specific to the linear encoders (DAP1)

Physical pulse resolution [nm]

WARNING
This parameter is active only if the **Scaling function control** parameter is set to "=0"; otherwise it is ignored and the system uses the custom values (**Programmable pulse resolution [nm]** and **Programmable Total resolution [pulse]**) to calculate the position information.
Furthermore, if the **Bypass** parameter (see on page 92) is set to "1" = enabled, the scaling function -even if enabled- is ignored and the position information is outputted as it is.

It sets the physical resolution of the linear encoder expressed in nanometres (nm). The value has to be comprised between 1 and 1,000,000 (1 mm). Usually the physical resolution can be read in the order code (see the product datasheet).
Default = 5,000 (min. = 1, max. = 1,000,000)

EXAMPLE
We need to connect the following linear encoder: **SMA5-GA-50**.
As you can see in the product datasheet, "50" in the order code means a 0.05 mm resolution = 50,000 nm resolution. Thus you have to set the value 50,000 in this parameter. For further information refer also to the "User's manual".

EXAMPLE
We need to connect the following linear encoder: **SMAX-BG-100**.
As you can see in the product datasheet, "100" in the order code means a 0.1 mm resolution = 100,000 nm resolution. Thus you have to set the value 100,000 in this parameter. For further information refer also to the "User's manual".

Programmable pulse resolution [nm]

WARNING
This is only available when **Class 4 functionality** and **Scaling function control** are enabled.
You can activate a new value next to the **Programmable pulse resolution [nm]** parameter only if **Scaling function control** = ENABLED; if **Scaling function control** = ENABLED the set resolution values are enabled and used by the encoder; on the contrary, if **Scaling function control** = DISABLED you are allowed to set new resolution values and they are accepted, however the

encoder still goes on using the physical values, NOT the new entered values, until you enable the **Scaling function control**.
Furthermore, if the **Bypass** parameter (see on page 92) is set to "1" = enabled, the scaling function -even if enabled- is ignored and the position information is outputted as it is.

If **Class 4 functionality** = DISABLED or **Scaling function control** = DISABLED, then the system uses the **Physical pulse resolution [nm]**.

If the scaling function is enabled (**Scaling function control** = 1), the following condition has to be met:

$$\text{Programmable Total resolution [pulse]} \leq k * 2^{\text{Physical Total resolution [bit]}}$$

where:

$$k = \frac{\text{Physical pulse resolution [nm]}}{\text{Programmable pulse resolution [nm]}} \leq 1$$

This parameter is used to set a custom resolution (otherwise referred to as measuring step) expressed in nanometres [nm].
The resolution can be defined as the smallest change in the underlying quantity that produces a response in the measurement, the response being the information that is provided to output.
The custom resolution value must be greater than or equal to the physical resolution of the connected encoder (≥ **Physical pulse resolution [nm]**).
We suggest setting a value that is a multiple of the physical resolution as set next to the **Physical pulse resolution [nm]** parameter not to cause a counting error, i.e. a jump in the position count when the sensor crosses the physical zero point (see the **WARNING** below).

Default = 5,000 (min. = 1, max. = 1,000,000)

WARNING
When you enable the scaling function (**Scaling function control** = 1), please enter scaled values next to the **Programmable pulse resolution [nm]** and **Programmable Total resolution [pulse]** parameters that are consistent with the physical values. In the case of inconsistent values, the system warns about the wrong parametrization and fault condition.

EXAMPLE

We need to connect the following linear encoder: **SMA5-GA-50**.
As you can see in the product datasheet, "50" in the order code means a **0.05 mm resolution** = 50,000 nanometres resolution. The user has to confirm this value in the **Physical pulse resolution [nm]** parameter; if the **Scaling function control** parameter is disabled the system uses the physical resolution to calculate the position information. After enabling the **Scaling function control** parameter the system uses the custom resolution set next to the **Programmable pulse resolution [nm]** parameter: it must be greater than or equal to 50,000.

EXAMPLE

We need to connect the following linear encoder: **SMAX-BG-100**.
As you can see in the product datasheet, "100" in the order code means a **0.1 mm resolution** = 100,000 nanometres resolution. The user has to confirm this value in the **Physical pulse resolution [nm]** parameter; if the **Scaling function control** parameter is disabled the system uses the physical resolution to calculate the position information. After enabling the **Scaling function control** parameter the system uses the custom resolution set next to the **Programmable pulse resolution [nm]** parameter: it must be greater than or equal to 100,000.

WARNING

When you enable the scaling function (**Scaling function control** = 1), a counting error, i.e. a jump in the position count, may occur if the following conditions arise:

- a physical zero setting has been performed in the linear sensor;
- the **Programmable pulse resolution [nm]** parameter value is not a multiple of the physical resolution as set next to the **Physical pulse resolution [nm]** parameter;
- the measuring range (**Programmable Total resolution [pulse]** parameter) is not a power of 2 submultiple of the maximum measuring range ($2^{\text{Physical Total resolution [bit]}}$).

If the above described conditions arise, a counting error may occur when the sensor crosses the physical zero point.
If the scaling function is disabled (**Scaling function control** = 0), the transmitted position values are always consistent.
If the scaling function is enabled (**Scaling function control** = 1) yet no physical zero setting has been performed in the linear sensor, the transmitted position values are always consistent.

If the scaling function is enabled (**Scaling function control** = 1), the **Programmable pulse resolution [nm]** parameter value is a multiple of the physical resolution and the measuring range (**Programmable Total resolution [pulse]**) is a power of 2 submultiple of the maximum measuring range ($2^{\text{Physical Total resolution [bit]}}$), the transmitted position values are consistent, regardless of the physical zero setting.

NOTE
If you have set and activated the preset, when you change the value next to the **Programmable pulse resolution [nm]** parameter, then you must check the value in the preset value and perform the homing operation.

EXAMPLE
The main and default features of the **SMAX-BG-100** linear encoder are as follows:

1. **Physical resolution** = **Physical pulse resolution [nm]** = 0.1 mm = 100,000 nm
2. **MTAX max. measuring length** = 600 mm
3. **Max. number of information** = 6,000 (13 bits)

The max. number of information provided to output is calculated as follows:

$$\text{Number of information} = \frac{\text{Max. measuring length}}{\text{Resolution}}$$

Thus, in a default configuration the number of information is:

$$\text{Number of information} = \frac{\text{Max. measuring length}}{\text{Physical resolution}} = \frac{600}{0.1} = 6{,}000$$

Let's assume that you need **2,000 information** to be provided to output for the max. measuring length. It follows that you need to calculate and then set a custom resolution.
The resolution value results from the following calculation:

$$\text{Resolution} = \frac{\text{Max. measuring length}}{\text{Number of information}}$$

Thus, in the example the resolution will be:

$$\text{Resolution} = \frac{\text{Max. measuring length}}{\text{Number of information}} = \frac{600}{2{,}000} = 0.3$$

As the value next to the **Programmable pulse resolution [nm]** parameter has to be expressed in nanometres, then you have to enter the value **300,000**.

The complete programming sequence will be:

1. Enable the **Scaling function control** = 1
2. Set the custom resolution: **Programmable pulse resolution [nm]** = 300,000
3. Set the custom number of information: **Programmable total measuring range (pulse)** = 2,000

NOTE

Please note that, if you set a preset along the path, when the encoder moves back and cross the zero, the value immediately after 0 will be 1,999 as shown below.

...	1,996	1,997	1,998	1,999	0	1	2	3	4	...

Physical Total resolution [bit]

WARNING

This parameter is active only if the **Scaling function control** parameter is set to "=0"; otherwise it is ignored and the system uses the custom values (**Programmable pulse resolution [nm]** and **Programmable Total resolution [pulse]**) to calculate the position information.
Furthermore, if the **Bypass** parameter (see on page 92) is set to "1" = enabled, the scaling function -even if enabled- is ignored and the position information is outputted as it is.

It sets the max. number of information (expressed in bits) the SSI encoder can output for the max. measuring length. The value depends on the encoder resolution and the max. measuring length and has to be comprised between 1

and 30. For any information on the max. number of information please refer to the "User's manual" of the connected encoder.
Default = 30 (min. = 2, max. = 30)

EXAMPLE
We need to connect the following linear encoder: **SMA5-GA-50**. Its resolution is 0.05 mm (see the order code).
The max. measuring length of the the SMA5 linear encoder on the MTA5 scale is 5,050 mm.
The max. number of information the encoder can output results from the following calculation:

$$\text{Total Physical Resolution} = \frac{\text{Max. measuring range}}{\text{Physical resolution}}$$

$$\text{Total Physical Resolution} = \frac{5{,}050}{0.05} = \mathbf{101{,}000}$$

Now you have to "round up" the result to the next highest power of 2, that is: 131,072 = 2^{17}. Thus the number of bits is "17". The value to set in this parameter is 17.

EXAMPLE
We need to connect the following linear encoder: **SMAX-BG-100**. Its resolution is 0.1 mm (see the order code).
The max. measuring length of the SMAX linear encoder on the MTAX scale is 600 mm.
The max. number of information the encoder can output results from the following calculation:

$$\text{Total Physical Resolution} = \frac{\text{Max. measuring range}}{\text{Physical resolution}}$$

$$\text{Total Physical Resolution} = \frac{600}{0.1} = \mathbf{6{,}000}$$

Now you have to "round up" the result to the next highest power of 2, that is: 8,192 = 2^{13}. Thus the number of bits is "13". The value to set in this parameter is 13.

Programmable Total resolution [pulse]

WARNING
This is only available when **Class 4 functionality** and **Scaling function control** are enabled.
You can activate a new value next to the **Programmable Total resolution [pulse]** parameter only if **Scaling function control** = ENABLED; if **Scaling function control** = ENABLED the set resolution values are enabled and used by the encoder; on the contrary, if **Scaling function control** = DISABLED you are allowed to set new resolution values and they are accepted, however the encoder still goes on using the physical values, NOT the new entered values, until you enable the **Scaling function control**.
Furthermore, if the **Bypass** parameter (see on page 92) is set to "1" = enabled, the scaling function -even if enabled- is ignored and the position information is outputted as it is.

If **Class 4 functionality** = DISABLED or **Scaling function control** = DISABLED, then the system uses the **Physical Total resolution [bit]**.

If the scaling function is enabled (**Scaling function control** = 1), the following condition has to be met:

$$\text{Programmable Total resolution [pulse]} \leq k * 2^{\text{Physical Total resolution [bit]}}$$

where:

$$k = \frac{\text{Physical pulse resolution [nm]}}{\text{Programmable pulse resolution [nm]}} \leq 1$$

It sets the number of information for the whole length of the travel the encoder has to measure. It has to be comprised between 1 and 2^{30} = 1,073,741,824.
It can be either the number of information for the max. measuring length (for instance, if the application needs the whole path); or the number of information for just a part of the scale if the application only uses a section of the scale. Thus this value must be lower than or equal to the number of information resulting from the max. measuring length of the scale ($2^{\text{Physical Total resolution [bit]}}$).
We suggest setting a value that is a power of 2 submultiple of the maximum measuring range (**Physical Total resolution [bit]**) not to cause a counting error, i.e. a jump in the position count when the sensor crosses the physical zero point (see the **WARNING** below).
Default = 1,073,741,824 (min. = 2, max. = 1,073,741,824)

WARNING

When you enable the scaling function (**Scaling function control** = 1), please enter scaled values next to the **Programmable pulse resolution [nm]** and **Programmable Total resolution [pulse]** parameters that are consistent with the physical values. In the case of inconsistent values, the system warns about the wrong parametrization and fault condition.

EXAMPLE

We need to connect the following linear encoder: **SMA5-GA-50**.
As you can see in the product datasheet, "50" in the order code means a 0.05 mm resolution. Let's say the mechanical travel of our application is the max. measuring length the SMA5 linear encoder is allowed to run on the MTA5 scale, i.e. 5,050 mm. Thus the max. number of information is 101,000 ≈ 17 bits (for the complete explanation refer to the **Programmable pulse resolution [nm]** parameter). If you need a custom measuring range, you need to enable the **Scaling function control** and then set a value lower than 2^{17} = 131,072 in this parameter.
If you set a preset along the path, when the encoder moves back and cross the zero, the value immediately after 0 will be **Programmable Total resolution [pulse]** (= $2^{\text{Physical Total resolution [bit]}}$ in the example) - 1, i.e. 131,071.

...	131,069	131,070	131,071	0	1	2	...

EXAMPLE

We need to connect the following linear encoder: **SMAX-BG-100**.
As you can see in the product datasheet, "100" in the order code means a 0.1 mm resolution. Let's say the mechanical travel of our application is the max. measuring length the SMAX linear encoder is allowed to run on the MTAX scale, i.e. 600 mm. Thus the max. number of information is 6,000 ≈ 13 bits (for the complete explanation refer to the **Programmable pulse resolution [nm]** parameter). If you need a custom measuring range, you need to enable the **Scaling function control** and then set a value lower than 2^{13} = 8,192 in this parameter.
If you set a preset along the path, when the encoder moves back and cross the zero, the value immediately after 0 will be **Programmable Total resolution [pulse]** (= $2^{\text{Physical Total resolution [bit]}}$ in the example) - 1, i.e. 8,191.

...	8,189	8,190	8,191	0	1	2	...

EXAMPLE

We need to connect an SMA5-GA-50, its physical resolution is 0.05 mm. Let's say the mechanical travel of our application is 1,000 mm. Thus the max. number of information is 20,000 ≈ 15 bits (for the complete explanation refer to the **Programmable pulse resolution [nm]** parameter). Thus you must enable the **Scaling function control** parameter and set the value 20,000 in this parameter (instead of the physical value 131,072).
In this way you will obtain several 20,000 information sections following each other all along the whole measuring length. The position information will be from 0 to 19,999; then again from 0 to 19,999 and so on.

...	19997	19998	19999	0	1	2	...	19997	19998	19999	0	1	2	...

← max measuring length →

WARNING

When you enable the scaling function (**Scaling function control** = 1), a counting error, i.e. a jump in the position count, may occur if the following conditions arise:

- a physical zero setting has been performed in the linear sensor;
- the **Programmable pulse resolution [nm]** parameter value is not a multiple of the physical resolution as set next to the **Physical pulse resolution [nm]** parameter;
- the measuring range (**Programmable Total resolution [pulse]** parameter) is not a power of 2 submultiple of the maximum measuring range ($2^{\text{Physical Total resolution [bit]}}$).

If the above described conditions arise, a counting error may occur when the sensor crosses the physical zero point.
If the scaling function is disabled (**Scaling function control** = 0), the transmitted position values are always consistent.
If the scaling function is enabled (**Scaling function control** = 1) yet no physical zero setting has been performed in the linear sensor, the transmitted position values are always consistent.
If the scaling function is enabled (**Scaling function control** = 1), the **Programmable pulse resolution [nm]** parameter value is a multiple of the physical resolution and the measuring range (**Programmable Total resolution [pulse]**) is a power of 2 submultiple of the maximum measuring range ($2^{\text{Physical Total resolution [bit]}}$), the transmitted position values are consistent, regardless of the physical zero setting.

WARNING

When you change the value next to **Programmable Total resolution [pulse]** parameter, then you must check the value the preset value and perform the preset operation.

9.4.6 Description of the parameters specific to the rotary encoders (DAP2)

Physical singleturn resolution [bit]

WARNING
This parameter is active only if the **Scaling function control** parameter is set to "=0"; otherwise it is ignored and the system uses the custom values (**Programmable pulse/rev** and **Programmable total measuring range (pulse)**) to calculate the position information.
Furthermore, if the **Bypass** parameter (see on page 92) is set to "1" = enabled, the scaling function -even if enabled- is ignored and the position information is outputted as it is.

It sets the physical singleturn resolution (the number of physical distinguishable steps per each revolution) of the SSI encoder expressed in bits.
The value has to be comprised between 1 and 18. Usually the physical resolution can be read in the order code (see the product datasheet).
Default = 16 (min. = 1, max. = 18)

EXAMPLE
We need to connect the following rotary encoder: **MM36 12/8192**.
As you can see in the product datasheet, "12" in the order code means a physical singleturn resolution of 12 bits (4,096 cpr). Thus you have to set the value 12 in this parameter. For further information refer also to the "User's manual" of the connected encoder.

EXAMPLE
We need to connect the following rotary encoder: **HM58 16/16384**.
As you can see in the product datasheet, "16" in the order code means a physical singleturn resolution of 16 bits (65,536 cpr). Thus you have to set the value 16 in this parameter. For further information refer also to the "User's manual" of the connected encoder.

Physical multiturn resolution [bit]

WARNING
This parameter is active only if the **Scaling function control** parameter is set to "=0"; otherwise it is ignored and the system uses the custom values (**Programmable pulse/rev** and **Programmable total measuring range (pulse)**) to calculate the position information.

Furthermore, if the **Bypass** parameter (see on page 92) is set to "1" = enabled, the scaling function -even if enabled- is ignored and the position information is outputted as it is.

It sets the physical multiturn resolution (the number of physical revolutions) of the SSI encoder expressed in bits.
The value has to be comprised between 0 and 14. Usually the physical resolution can be read in the order code (see the product datasheet).
Default = 14 (min. = 0, max. = 14)

EXAMPLE
We need to connect the following rotary encoder: **AS58 13**.
AS58 is a singleturn encoder, thus its physical number of revolutions is 1. To translate the number of revolutions into bits, you must calculate the power of 2 of the value: $1 = 2^0$. Thus the value to be set in this parameter is 0. For further information refer also to the "User's manual" of the connected encoder.

EXAMPLE
We need to connect the following rotary encoder: **MM36 12/8192**.
In the order code, the hardware multiturn resolution is usually expressed in number of revolutions. To translate the number of revolutions into bits, you must calculate the power of 2 of the value: $8,192 = 2^{13}$. Thus the value to be set in this parameter is 13. For further information refer also to the "User's manual" of the connected encoder.

EXAMPLE
We need to connect the following rotary encoder: **HM58 16/16384**.
In the order code, the hardware multiturn resolution is usually expressed in number of revolutions. To translate the number of revolutions into bits, you must calculate the power of 2 of the value: $16,384 = 2^{14}$. Thus the value to be set in this parameter is 14. For further information refer also to the "User's manual" of the connected encoder.

Programmable pulse/rev

WARNING
This is only available when **Class 4 functionality** and **Scaling function control** are enabled.
You can activate a new value next to the **Programmable pulse/rev** parameter only if **Scaling function control** = ENABLED; if **Scaling function control** = ENABLED the set resolution values are enabled and used by the encoder; on the

contrary, if **Scaling function control** = DISABLED you are allowed to set new resolution values and they are accepted, however the encoder still goes on using the physical values, NOT the new entered values, until you enable the **Scaling function control**.
Furthermore, if the **Bypass** parameter (see on page 92) is set to "1" = enabled, the scaling function -even if enabled- is ignored and the position information is outputted as it is.

If **Class 4 functionality** = DISABLED or **Scaling function control** = DISABLED, then the system uses the **Physical singleturn resolution [bit]**.

If the scaling function is enabled (**Scaling function control** = 1), the following condition has to be met:

$$\text{Programmable total measuring range (pulse)} \leq k * 2^{\text{Total physical resolution [bit]}}$$ (= Physical singleturn resolution [bit] * Physical multiturn resolution [bit])

where:

$$k = \frac{\text{Programmable pulse/rev}}{\text{Physical singleturn resolution [bit]}} \leq 1$$

This parameter is used to set a custom number of distinguishable steps per revolution (singleturn resolution).
The custom singleturn resolution value must be less than or equal to the physical singleturn resolution of the connected encoder:

$$\text{Programmable pulse/rev} \leq 2^{\text{Physical singleturn resolution [bit]}}$$

You are allowed to set any integer value less than or equal to the Hardware counts per revolution. However we suggest setting a value that is a power of 2. This is meant to avoid counting errors.

Default = 65,536 (min. = 1, max. = 262,144)

WARNING
When you set a new value next to the **Programmable pulse/rev** item, please always check also the **Programmable total measuring range (pulse)** item value and be sure that the resulting number of revolutions complies with the Hardware number of revolutions of the device.

Let's suppose that our HM58 16/16384 encoder is programmed as follows:
Programmable pulse/rev: 8,192
Programmable total measuring range (pulse) = $33{,}554{,}432_{10}$ = 8,192 (cpr) * 4,096 (rev.)
Let's set a new singleturn resolution, for instance: **Programmable pulse/rev** = 360.

If we do not change the **Programmable total measuring range (pulse)** value at the same time, we will get the following result:

$$\text{Number of revolutions} = \frac{33{,}554{,}432\ (\textbf{Programmable total measuring range (pulse)})}{360\ (\textbf{Programmable pulse/rev})} = 93{,}206.755...$$

As you can see, the encoder is required to carry out more than 93,000 revolutions, this cannot be as the hardware number of revolutions is, as stated, 16,384.

WARNING
When you enable the scaling function (**Scaling function control** = 1), please enter scaled values next to the **Programmable pulse/rev** and **Programmable total measuring range (pulse)** parameters that are consistent with the physical values. In the case of inconsistent values, the system warns about the wrong parametrization and fault condition.

WARNING
Every time you change the value in this parameter then you are required to set a new preset value (see the **G1_XIST1 Preset control** parameter).

Programmable total measuring range (pulse)

WARNING
This is only available when **Class 4 functionality** and **Scaling function control** are enabled.
You can activate a new value next to the **Programmable total measuring range (pulse)** parameter only if **Scaling function control** = ENABLED; if **Scaling function control** = ENABLED the set resolution values are enabled and used by the encoder; on the contrary, if **Scaling function control** = DISABLED

you are allowed to set new resolution values and they are accepted, however the encoder still goes on using the physical values, NOT the new entered values, until you enable the **Scaling function control**.
Furthermore, if the **Bypass** parameter (see on page 92) is set to "1" = enabled, the scaling function -even if enabled- is ignored and the position information is outputted as it is.

If **Class 4 functionality** = DISABLED or **Scaling function control** = DISABLED, then the system uses the Total physical resolution (= $2^{\text{Physical singleturn resolution [bit]} + \text{Physical multiturn resolution [bit]}}$).

If the scaling function is enabled (**Scaling function control** = 1), the following condition has to be met:

$$\text{Programmable total measuring range (pulse)} \leq k * 2^{\text{Total physical resolution [bit]}} (= \text{Physical singleturn resolution [bit]} * \text{Physical multiturn resolution [bit]})$$

where:

$$k = \frac{\text{Programmable pulse/rev}}{\text{Physical singleturn resolution [bit]}} \leq 1$$

This parameter sets a custom number of distinguishable steps over the total measuring range. The **Programmable total measuring range (pulse)** (total resolution of the encoder) results from the product of **Programmable pulse/rev** by the required Number of revolutions: **Programmable total measuring range (pulse)** = **Programmable pulse/rev** * Number of revolutions.

Allowed values are less than or equal to the Total physical resolution of the connected encoder:

Programmable total measuring range (pulse) ≤ Total physical resolution

We recommend the **Programmable total measuring range (pulse)** to be set to a power of 2. This is meant to avoid counting errors.

Default = 1,073,741,824 (min. = 1, max. = 1,073,741,824)

WARNING

When you set a new value next to the **Programmable total measuring range (pulse)** item, please always check also the **Programmable pulse/rev** item value and be sure that the resulting number of revolutions complies with the Hardware number of revolutions (**Physical multiturn resolution [bit]**) of the device.

Let's suppose that our HM58 16/16384 encoder is programmed as follows:

Programmable pulse/rev: 8192

Programmable total measuring range (pulse) = $33{,}554{,}432_{10}$ = 8192 (cpr) * 4096 (rev.)

Let's set a new total resolution, for instance: **Programmable total measuring range (pulse)** = 360.

As the **Programmable total measuring range (pulse)** must be greater than or equal to the **Programmable pulse/rev**, the above setting is not allowed.

WARNING

When you enable the scaling function (**Scaling function control** = 1), please enter scaled values next to the **Programmable pulse/rev** and **Programmable total measuring range (pulse)** parameters that are consistent with the physical values. In the case of inconsistent values, the system warns about the wrong parametrization and fault condition.

WARNING

Every time you change the value in this parameter then you are required to set a new preset value (see the **G1_XIST1 Preset control** parameter).

EXAMPLE

We connect the HM58 16/16384 rotary encoder.

The physical resolution is as follows:

- Hardware counts per revolution: 65,536 cpr = 16 bits; **Physical singleturn resolution [bit]** = 16
- Hardware number of revolutions: 16,384 rev. = 14 bits; **Physical multiturn resolution [bit]** = 14
- Total hardware resolution: **Physical singleturn resolution [bit]** + **Physical multiturn resolution [bit]** = 16 + 14 = 30 bits (2^{30} = 1,073,741,824 information)

In the specific installation 2,048 counts/rev. * 1,024 revolutions are required:

- enable the Class 4 functionality: **Class 4 functionality** = 1 = ENABLED

- enable the scaling function: **Scaling function control** = 1 = ENABLED
- set the counts per revolution: **Programmable pulse/rev** = 2,048
- set the total resolution: **Programmable total measuring range (pulse)** = **Programmable pulse/rev** * Custom number of revolutions = 2,048 * 1,024 = 2,097,152

NOTE
We suggest setting values which are power of 2 (2^n: 1, 2, 4, ..., 2,048, 4,096, 8,192, ...) next to the **Programmable pulse/rev** and **Programmable total measuring range (pulse)** parameters to avoid counting errors.

NOTE
Any multiturn encoder can be configured so that it works exactly as a singleturn encoder. This is achieved by setting **Programmable total measuring range (pulse)** = **Programmable pulse/rev** (furthermore the **Physical multiturn resolution [bit]** has to be set to 0). Let's suppose the encoder is set as follows:
Programmable pulse/rev = 8,192
Programmable total measuring range (pulse) = 8,192

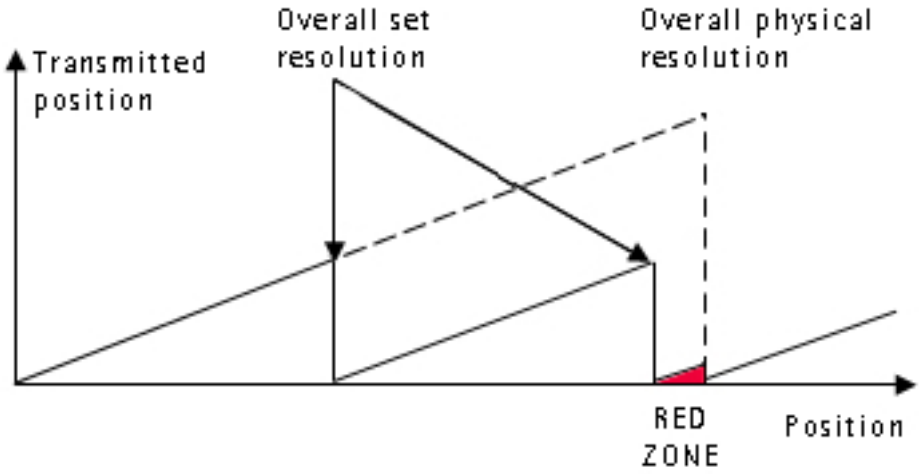
So it follows that:

$$\text{Number of revolutions} = \frac{8{,}192\ (\textbf{Programmable total measuring range (pulse)})}{8{,}192\ (\textbf{Programmable pulse/rev})} = 1$$

This is exactly the configuration of the singleturn encoder.
Of course the contrary is not possible.

9.5 "Red Zone" (rotary encoders only)

The so-called "Red Zone" problem occurs when the **Number of revolutions** (i.e. the **Programmable total measuring range (pulse)** / **Programmable pulse/rev**) is not a power of 2.
When this problem arises, the device must operate within the "red zone" for a certain number of positions. The dimension of the "red zone" is variable. To calculate it we must subtract the overall set resolution from the overall physical resolution of the device as many times as until the difference is less than the overall set resolution. When the encoder crosses the limit of the last value in the overall physical resolution, a counting error occurs, i.e. a jump in the position count. The problem is represented graphically in the following Figure.



EXAMPLE
HM58 16/16384 GA multiturn encoder

Physical resolution:

- **Physical singleturn resolution [bit]** = 65,536 counts/rev. = 16 bits (2^{16})
- **Physical multiturn resolution [bit]** = 16,384 revolutions = 14 bits (2^{14})
- Overall physical resolution = 1,073,741,824 = 30 bits (2^{30})

Set values:

- **Programmable pulse/rev** = 65,536 = 2^{16}
- **Programmable total measuring range (pulse)** = 442,236,928 = it is NOT a power of 2

In fact:

- **Number of revolutions** = **Programmable total measuring range (pulse)** / **Programmable pulse/rev** = 6,748 = it is NOT a power of 2

This can be proved easily:

$$\frac{\textbf{Overall physical resolution}}{\textbf{Overall set resolution}} = \frac{1{,}073{,}741{,}824}{442{,}236{,}928} = 2{,}427\ldots$$

It follows that for 189,267,968 positions (1,073,741,824 – 442,236,928 * 2 = 189,267,968), i.e. for 2,888 revolutions, the encoder will work within the limits of the so-called "red zone". After position 189,267,968 (i.e. at the end of the "red zone") a position error (namely, a "jump" in the position count) would happen as the following position would be "0". See the Figure in the previous page.

NOTE

Make attention using the values sent by the encoder while working within the limits of the "Red Zone". When the encoder changes from normal status to "Red Zone" status (and vice versa) a jump of position occurs.

10 Diagnostics and Alarms

Diagnostics data is always transferred acyclically using Record Data communication over the non real time channel. A PN-IO controller can request diagnostic data from the PN-IO device using RDO (Record Data Object) services.

Alarm data is transmitted from the IO device to the IO controller via the RT channel.

The encoder errors are divided into **Faults** and **Warnings**, they are defined as follows.

FAULT
A Fault is set if a malfunction in the encoder could lead to incorrect position values.

WARNING
Warnings indicate that the tolerance for certain internal parameters of the encoder has been exceeded. Unlike faults, warnings do not imply incorrect position values.

NOTE
Please note that warnings are not supported in this encoder.

There are several diagnosis mechanisms that are used to monitor encoder diagnostics.
Please refer to the table below for an overview of the available diagnosis mechanisms.

Function	Reference
Acyclic diagnosis parameter **P65001 – Operating status**	See on page 115
Channel related diagnosis via the Alarm Channel	See on page 115
Error codes in **G1_XIST2**	See on page 116
LED indication	See on page 116

10.1 Acyclic diagnosis parameter P65001 – Operating status

With the Acyclic parameter **P65001 – Operating status** the current status of the Encoder Faults and Warnings as well as the support of the individual Fault and Warning bits can be read from the encoder. For detailed information on the parameter **P65001 – Operating status** please refer to page 84.

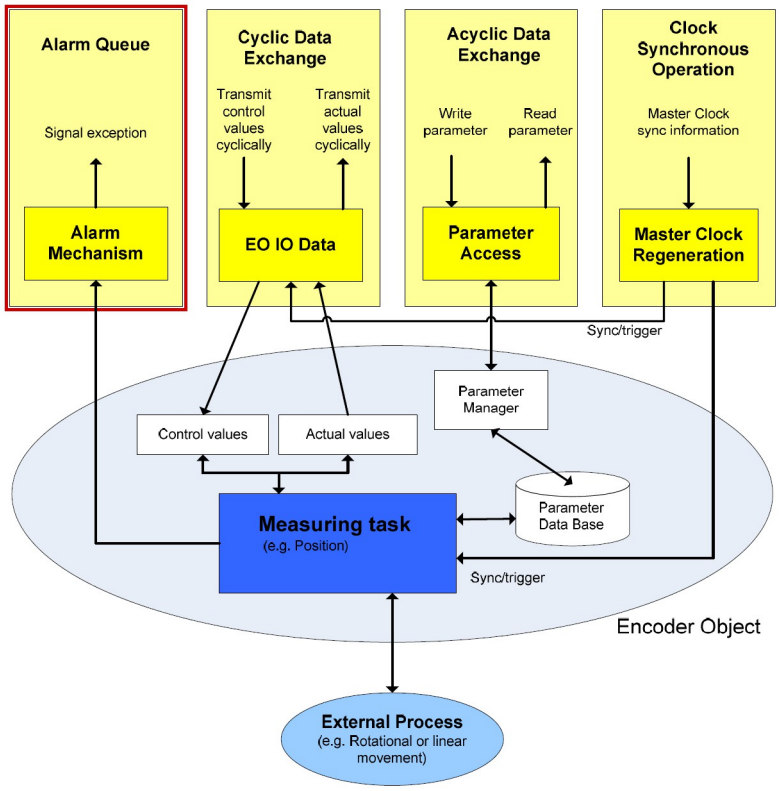
10.2 Error messages via the Alarm Channel

The encoder diagnosis is reported to the controller via the Alarm channel as Channel Related Diagnosis. Both warnings and faults are reported in the same manner but with different error types.

NOTE

In compatibility mode (see the **Compatibility Mode** parameter on page 90), channel related diagnostics can be switched off by the **Alarm channel control** parameter, please refer to its description on page 90.

For a detailed definition of the use of the channel related diagnosis please refer to the Alarm mechanism definition for the respective communication system in the mapping part of the profile.



10.2.1 Use of the ChannelErrorType

For Profinet the encoder faults and warnings are mapped to the ChannelErrorTypes defined in the PROFIdrive profile, see the tables below. This means that there are no specific codes defined for standalone encoders and a PROFINET controller will interpret the errors from an encoder in the same ways as an error coming from a drive. Refer also to the parameter **P65001 – Operating status** on page 84.

Error type	Definition	Explanation
0x9000	Memory error	The encoder failed to read saved offset or preset values from the internal non volatile memory.
0x9011	Commissioning diagnostic	User parameter data assignment error.

10.3 Error codes in G1_XIST2

Error codes are sent in **G1_XIST2** if an error occurs. For information about **G1_XIST2** refer to page 73.

G1_XIST2	Meaning	Explanation
0x0F02	Master's sign of life fault	The number of permissible failures of the master's sign of life was exceeded.
0x0F04	Synchronization fault	The number of permissible failures for the bus cycle was exceeded.
0x1001	Memory error	Error while writing on or reading the internal non volatile memory.
0x1002	Parametrization error	User parameter data assignment error. Example: Programmable pulse/rev and Programmable total measuring range (pulse) not compatible.

10.4 LED indication

Errors are further indicated through LEDs. Six LEDs located in the cap of the encoder (see Figure 4) are meant to show visually the operating or fault status of the encoder and the Profinet interface.
For detailed information refer to the "4.8 Diagnostic LEDs (Figure 4)" section on page 34.

11 Real time class communication

Within PROFINET IO, process data and alarms are always transmitted in real time. Real-Time for PROFINET (RT) is based on the definitions of IEEE and IEC for high-performance data exchange of I/O data. RT communication constitutes the basis for data exchange in PROFINET IO.

Real-time data are handled with higher priority compared to TCP(UDP)/IP data. This method of data exchange allows bus cycle times in the range of a few hundred milliseconds to be achieved.

Isochronous data exchange with PROFINET is defined in the Isochronous-Real-Time (IRT) concept. IRT communication is always clock synchronized and only possible within an IRT domain. Isochronous real-time communication differs from real-time communication mainly in its isochronous behaviour: the start of a bus cycle can deviate by a maximum of 1 µs (jitter is less then 1 µs). IRT is required in motion control applications (positioning operations), for example. This communication is required, for example, for high-accuracy closed-loop control tasks.

11.1 Real-time classes in PROFINET IO

To enable enhanced scaling of communication options and, thus, also of determinism in PROFINET IO, real-time classes have been defined for data exchange. From the user perspective, these classes involve unsynchronized and synchronized communication.
PROFINET IO differentiates the following classes for RT communication.
They differ not in terms of performance but in determinism.

11.2 Real-Time class 2 (RT2) – Not synchronized

In real-time class 2 frames are transmitted via unsynchronized communication (anisochronous communication).
To activate the real-time class 2 both the IO controller and the IO device must be set exactly the same as "Not synchronized". To do this proceed as follows.

11.2.1 Setting an anisochronous communication

To configure the IO controller (Figure 16) double click the **PN-IO** slot X2 to open the PN-IO properties dialog box. The **Properties – PN-IO** property sheet will appear. Enter the **Synchronization** tabbed page and select the **Not synchronized** value in the drop-down menu of the **Synchronization role** item. Confirm pressing the **OK** button.

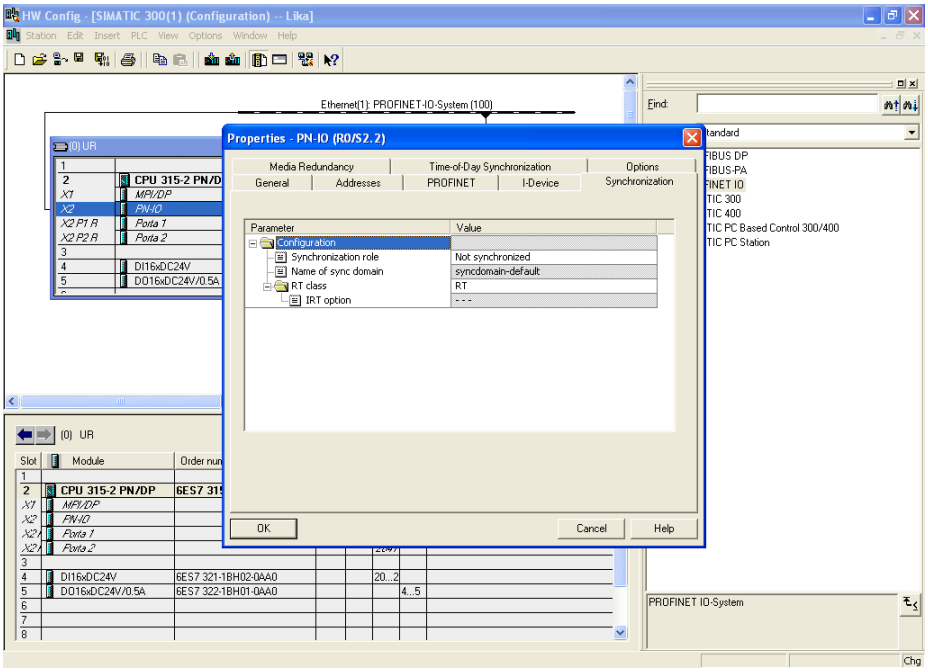


Figure 16 - Setting the Not synchronized role of the IO controller

To configure the IO device (Figure 17) double click the **Interface** slot X1 of the module to open the interface properties dialog box. The **Properties – Interface** property sheet will appear. Enter the **Synchronization** tabbed page and select the **Not synchronized** value in the drop-down menu of the **Synchronization role** item. Confirm pressing the **OK** button.

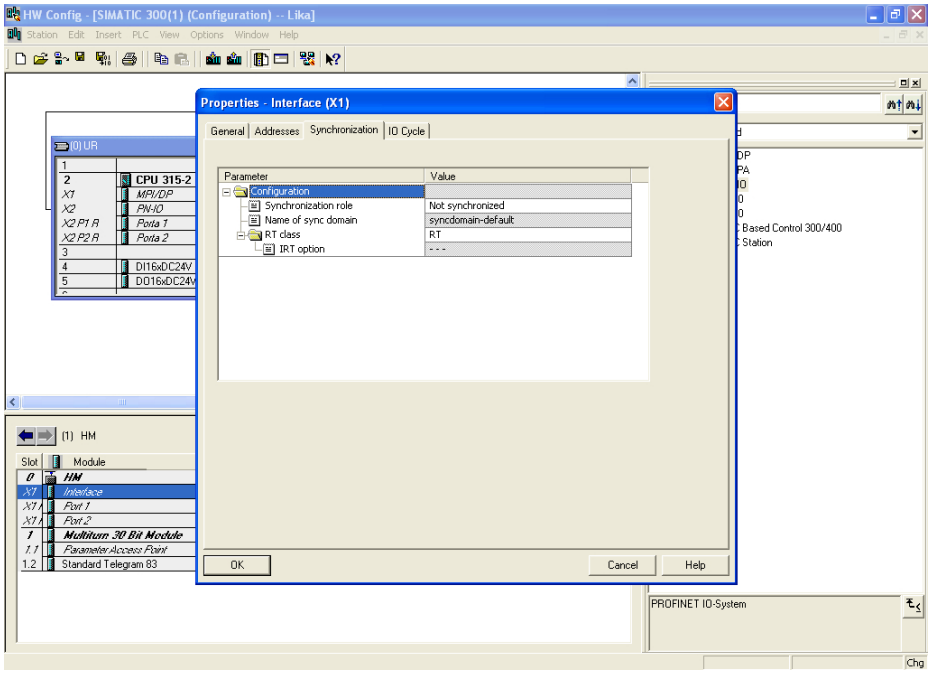


Figure 17 - Setting the Not synchronized role of the IO device

11.3 Real-Time class 3 (IRT_TOP) (RT3)

Isochronous data exchange with PROFINET is defined in the Isochronous-Real-Time (IRT) concept. IRT communication is always clock synchronized and only possible within an IRT domain. Isochronous real-time communication differs from real-time communication mainly in its isochronous behaviour: the start of a bus cycle can deviate by a maximum of 1 µs (jitter is less then 1 µs).
This communication is required, for example, for high-accuracy closed-loop control tasks.
Only industrial IRT switches can be used.
Typical cycle time 1 ms or less. All network components must support PROFINET IRT frame priority processing. Position values are captured with an accuracy of +/- 1 µs or better, with respect to the highly accurate bus clock.

11.3.1 Setting an isochronous communication

To activate the real-time class 3 both the IO controller and the IO device must be configured. To do this proceed as follows.

To configure the IO controller (Figure 18) double click the **PN-IO** slot X2 to open the PN-IO properties dialog box. The **Properties – PN-IO** property sheet will appear. Enter the **Synchronization** tabbed page and select the **Sync master** value in the drop-down menu of the **Synchronization role** item; select the **High performance** value in the drop-down menu of the **IRT option** item. Confirm pressing the **OK** button.

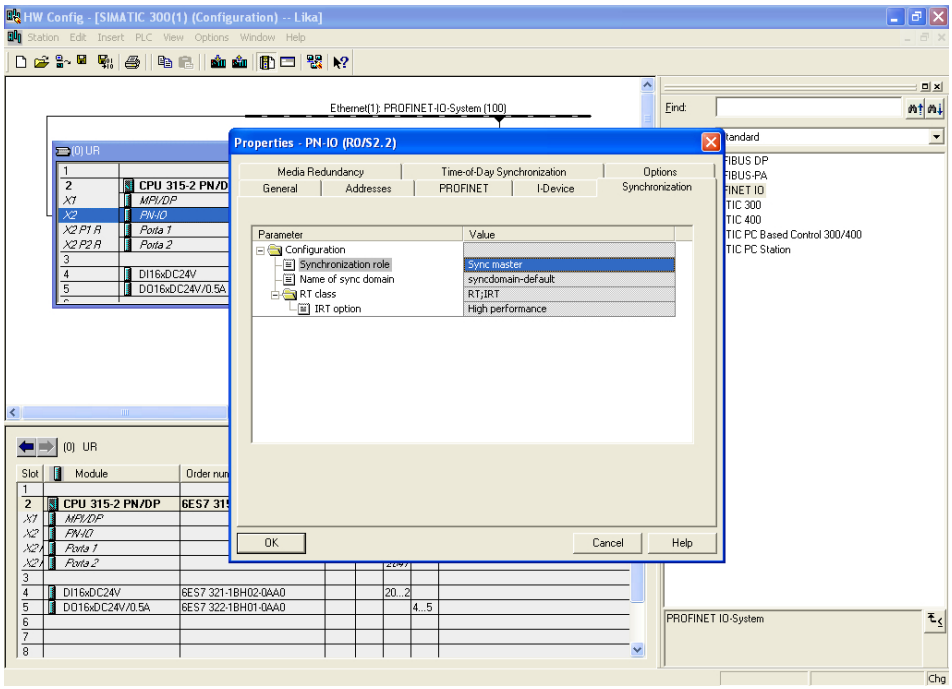


Figure 18 - Setting the sync master role of the IO controller

To configure the IO device (Figure 17) double click the **Interface** slot X1 of the module to open the interface properties dialog box. The **Properties – Interface** property sheet will appear. Enter the **Synchronization** tabbed page and select the **Sync slave** value in the drop-down menu of the **Synchronization role** item; select the **High performance** value in the drop-down menu of the **IRT option** item. Confirm pressing the **OK** button.

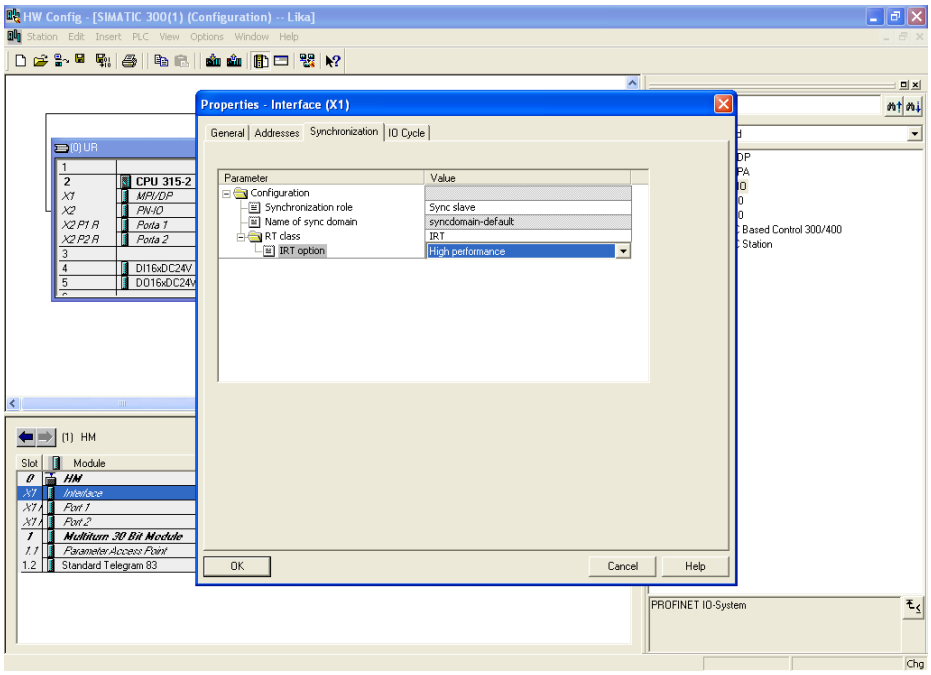


Figure 19 - Setting the sync slave role of the IO device

Now double click the **CPU** slot 2 in the CPU configuration table to open the CPU properties dialog box. The **Properties – CPU 315 ...** property sheet will appear. Enter the **Synchronous Cycle Interrupts** tabbed page and press the **Details** button.

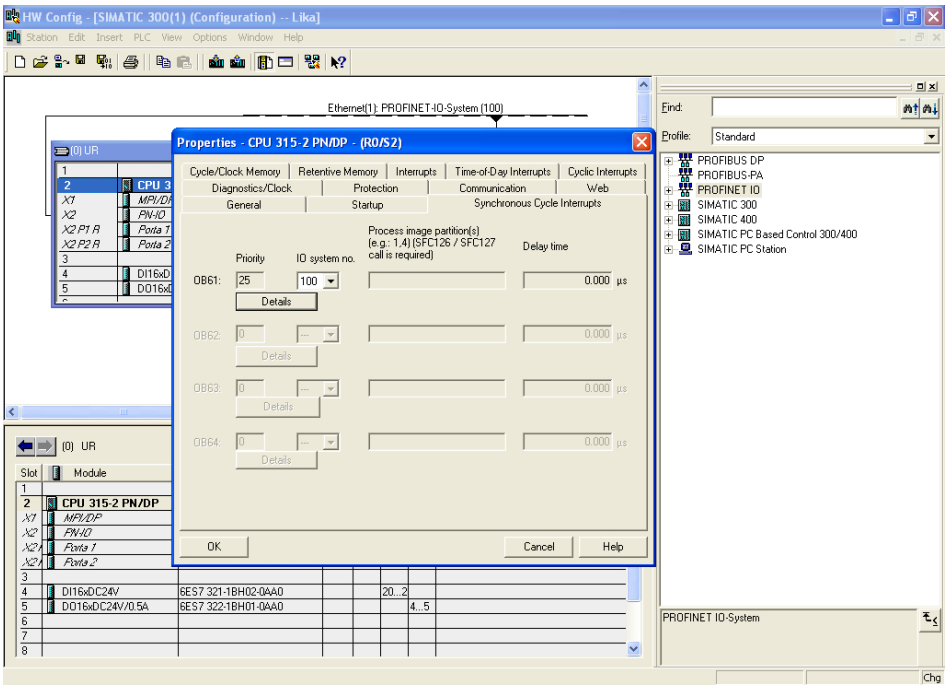


Figure 20 - CPU property sheet

The **OB61** dialog box will appear on the screen. Set the value "1" next to the **Process image partition(s)** item. Confirm pressing the **OK** button.

NOTE

For more information on PIPs (Process Image Partitions) refer to the "11.5 PIP (Process Image Partition)" section on page 132.

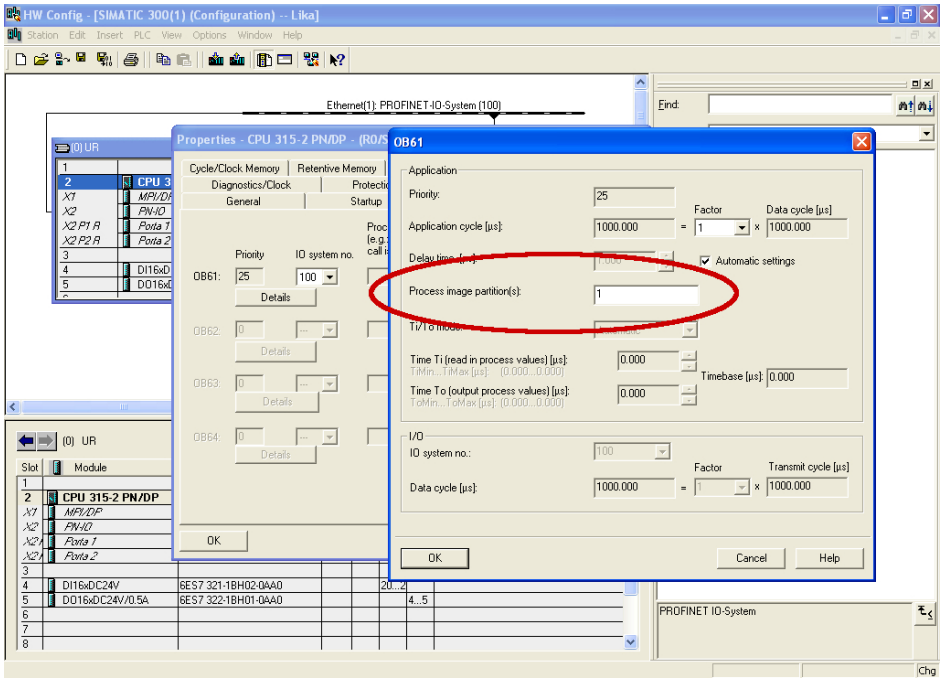


Figure 21 - PIP of OB61

Now set the encoder in isochronous mode. To do this double click the **Interface** slot X1 of the module to open the interface properties dialog box. The **Properties – Interface** property sheet will appear. Enter the **IO cycle** tabbed page and select the **OB61** value in the drop-down menu of the **Assign IO device in isochronous mode** item. For further information see also the "11.4 OB61" section on page 131.

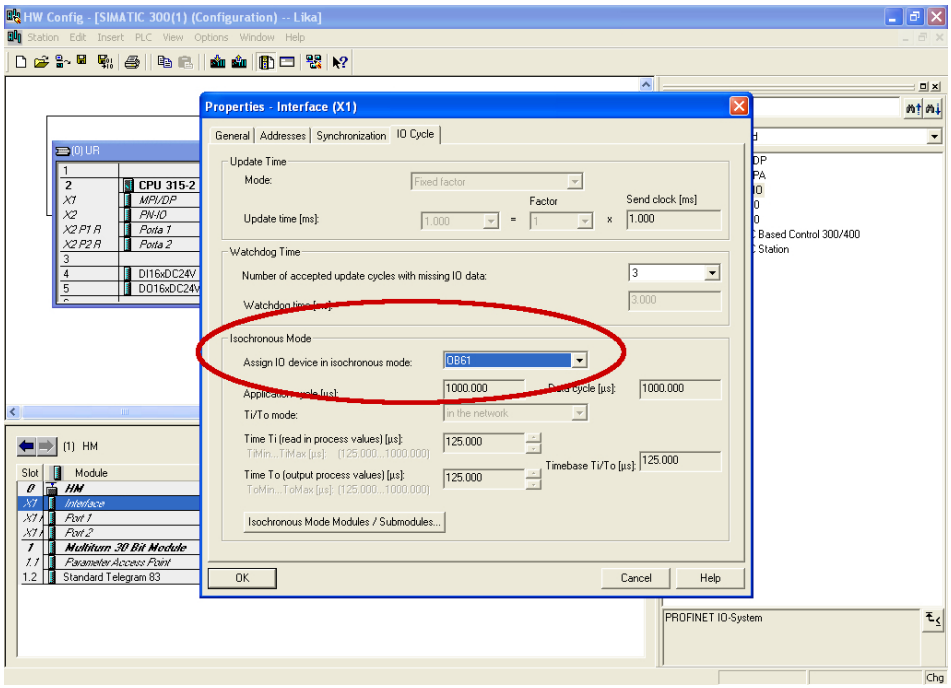


Figure 22 - OB61: assigning IO device in isochronous mode

Press the **Isochronous Mode Modules / Submodules ...** button below in the **Properties – Interface** property sheet and enter the **Isochronous Modules / Submodules** dialog box. Check that an isochronous submodule has been installed (Standard Telegram 81 in the example). Confirm pressing the **OK** button.

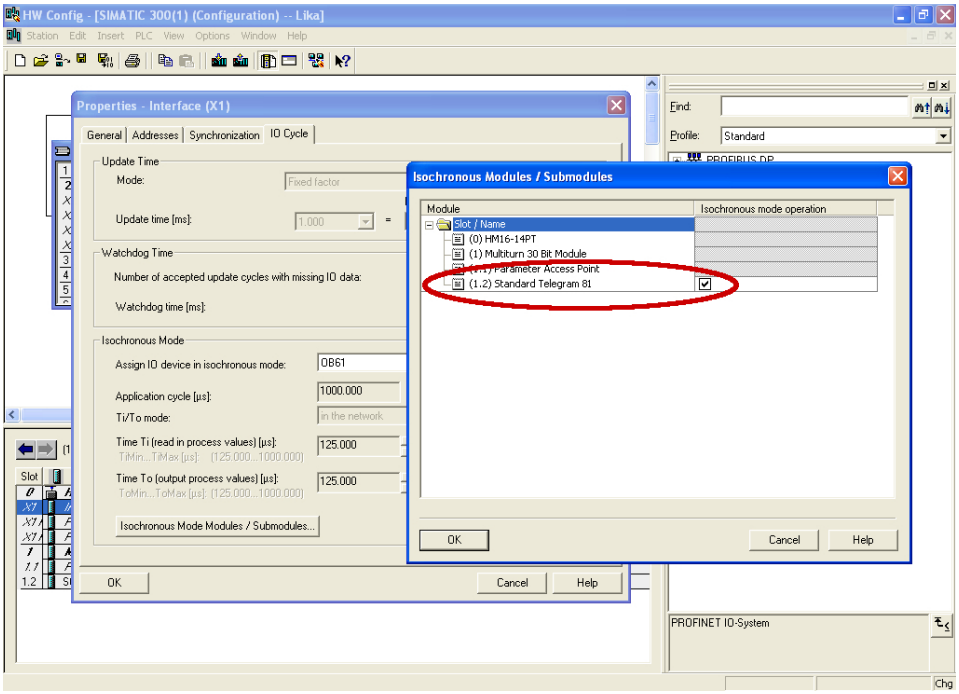


Figure 23 - Standard Telegram as isochronous submodule

Double click the **Standard Telegram** slot 1.2 of the module to open the telegrams properties dialog box. The **Properties – Standard Telegram** property sheet will appear. Enter the **Addresses** tabbed page and set the process image of the isochronous submodule as **PIP 1**. Select the **PIP1** value in the drop-down menu of the **Process image** items in both **Inputs** and **Outputs** group boxes. Confirm pressing the **OK** button.

NOTE

For more information on PIPs (Process Image Partitions) refer to the section "11.5 PIP (Process Image Partition)" on page 132.

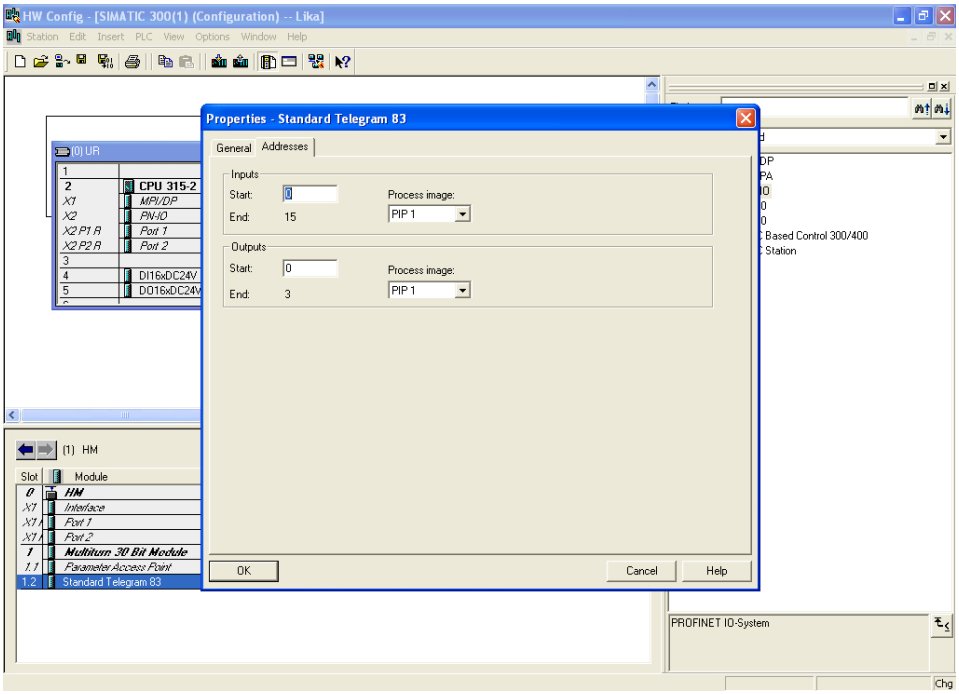


Figure 24 - Setting PIP for Standard Telegram IOs

Now you need to set a connection between the port of the controller and the port of the encoder. Please check the specific configuration of your Profinet network. In the example the port 2 of the IO controller is connected to the port 1 of the IO device; see Figure 37.
So, first configure the IO controller and set its port 2 to be connected to the port 1 of the IO device. Double click the **X2 P2 R PORT 2** slot in the CPU configuration table and open the Port 2 properties dialog box. The **Properties – PN-IO – Port 2** property sheet will appear. Enter the **Topology** tabbed page and set the **Port 1 (X1 P1)** value in the drop-down menu in the **Partner port** item of the **Partners** group box. Confirm pressing the **OK** button.

Refer also to the "11.7 Topology Editor" section on page 138.

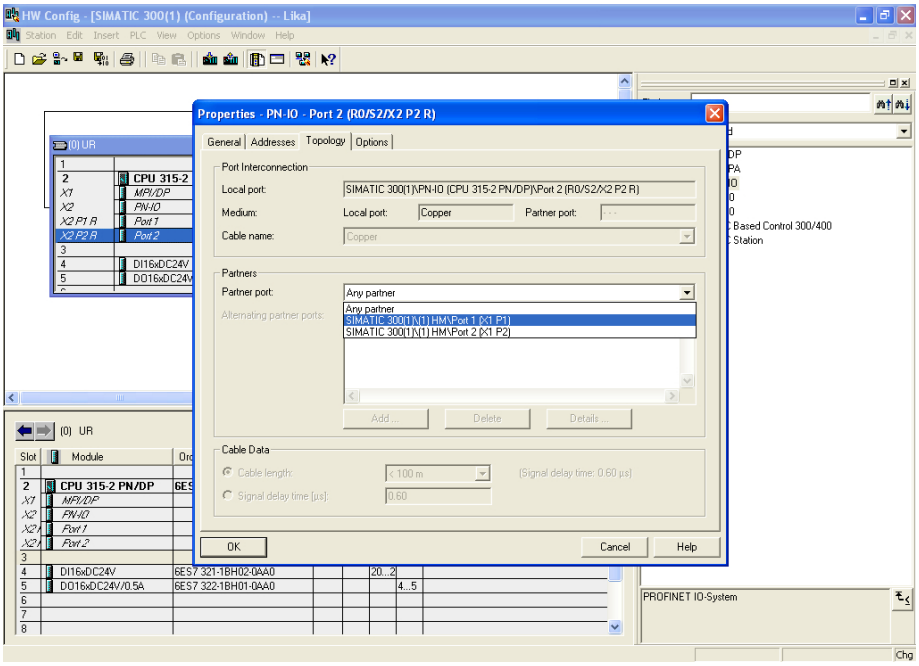


Figure 25- Configuring the IO controller topology

Then configure the IO device and set its port 1 to be connected to the port 2 of the IO controller. Double click the **X1 P1 PORT 1** slot in the Device configuration table and open the Port 1 properties dialog box. The **Properties – Interface – Port 1** property sheet will appear. Enter the **Topology** tabbed page and set the **Port 2 (X2 P2 R)** value in the drop-down menu in the **Partner port** item of the **Partners** group box. Confirm pressing the **OK** button.

Refer also to the "11.7 Topology Editor" section on page 138.

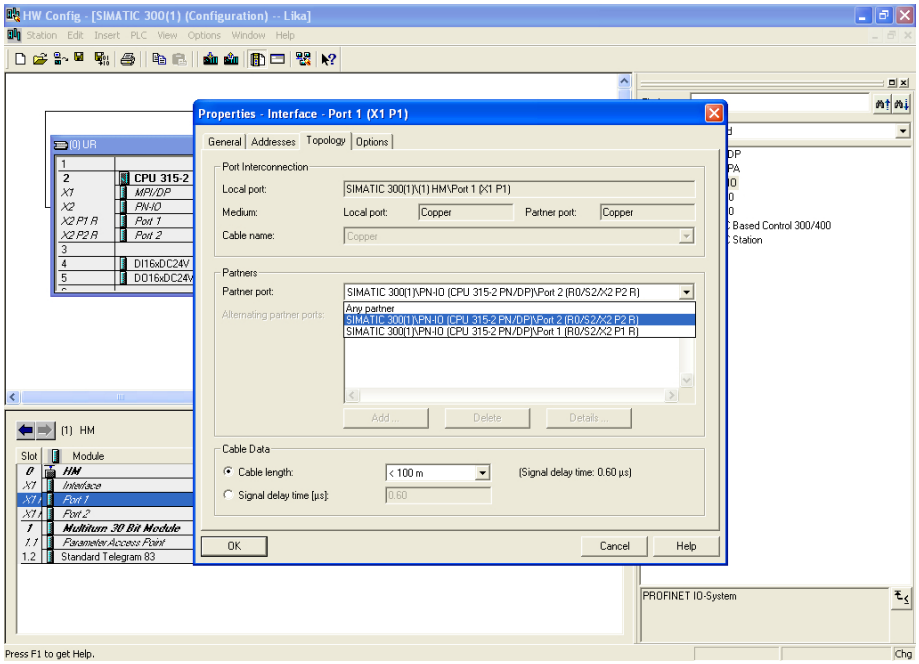


Figure 26 - Configuring the IO device topology

Finally check the Profinet IO isochronous mode.
Right click the **2 CPU** slot in the CPU configuration table and press the **PROFINET IO Isochronous mode** command.

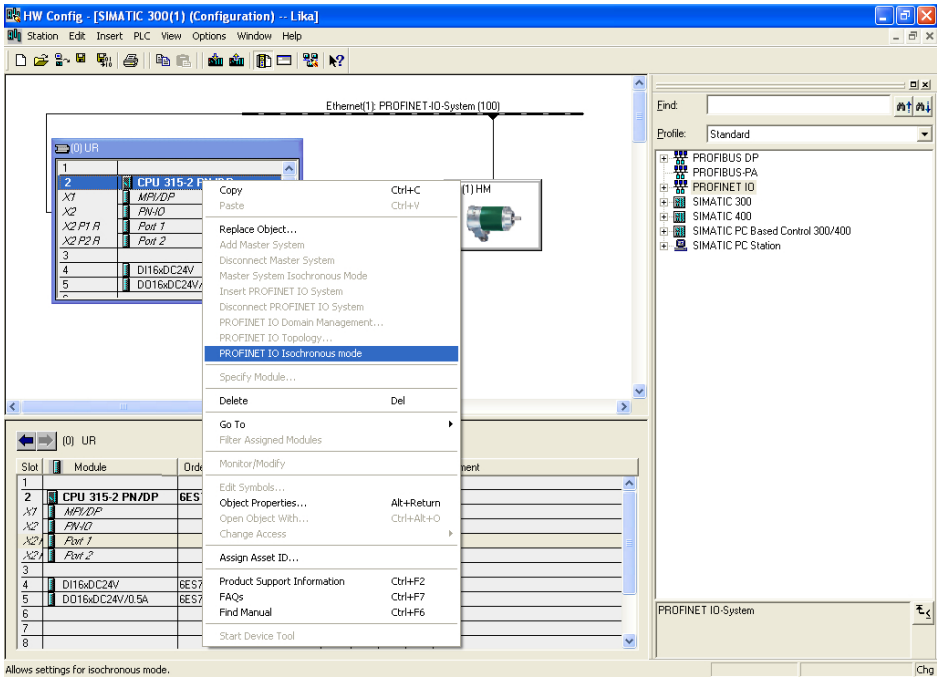


Figure 27 - Checking the Profinet IO isochronous mode

The **Isochronous Mode** dialog box will appear on the screen. The installed Standard Telegram is the only submodule which provides IO data.
In this page you can check which IO devices / modules / submodules have been set in isochronous mode with the IO controller.

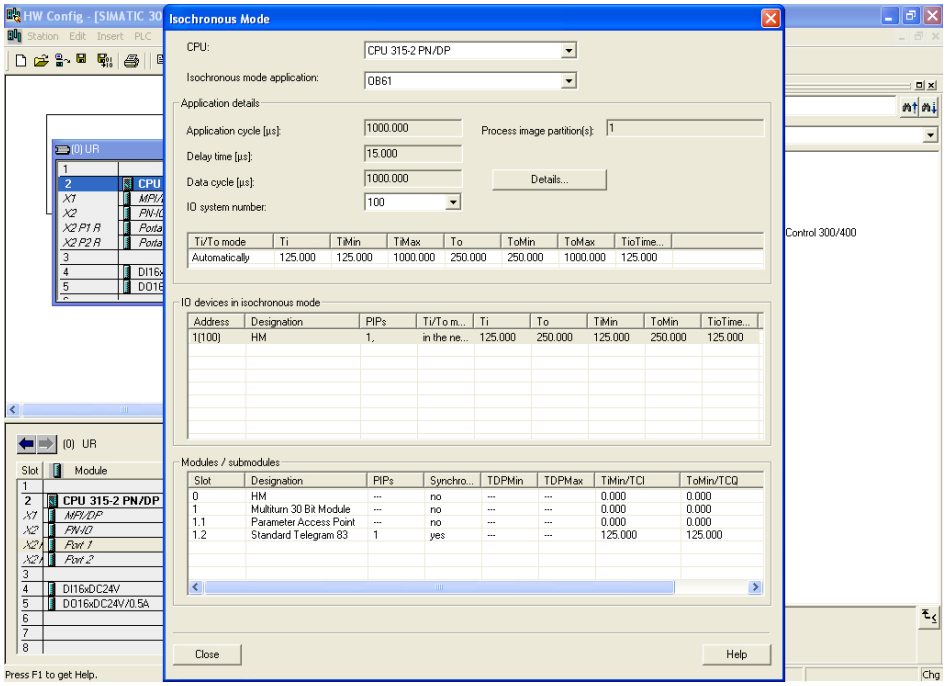


Figure 28 - Isochronous Mode dialog box

11.4 OB61

WARNING
Use of OBs requires both in-depth skills and specific expertise in SIMATIC STEP 7 programming environment. For detailed information please consult the STEP 7 Programmer's handbook and documentation.

Organization blocks (OBs) form the interface between the CPU operating system and the user program. The order in which the user program is executed is defined in the organization blocks.

The synchronization with the user program is maintained through the clocked interrupt OB61. OB61 is a synchronous cycle interrupt; in other words it is an isochronous event that is called with the start of every PROFINET cycle. It is synchronous with the Profinet send clock.

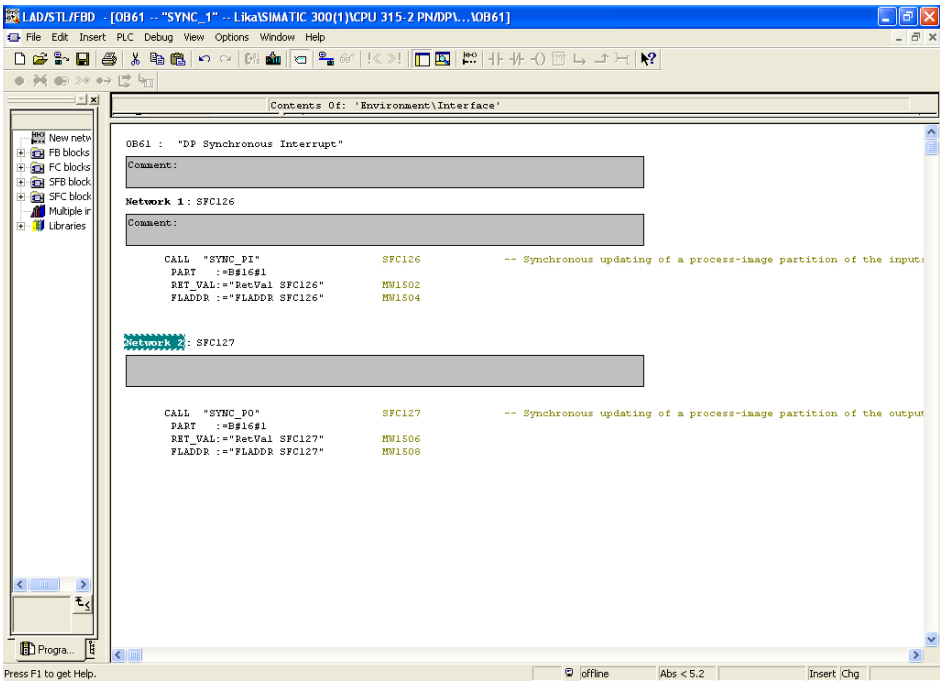


Figure 29 - OB61

11.5 PIP (Process Image Partition)

WARNING
Use of PIPs requires both in-depth skills and specific expertise in SIMATIC STEP 7 programming environment. For detailed information please consult the STEP 7 Programmer's handbook and documentation.

11.5.1 Consistency

PIPs (Process Image Partitions) are used to update the distributed IO device synchronously with the constant bus cycle time clock.
Compared with direct access to the input/output modules, the main advantage of accessing the process image is that the CPU has a consistent image of the process signals for the duration of one program cycle. If a signal state on an input module changes while the program is being executed, the signal state in the process image is retained until the process image is updated again in the next cycle. The process of repeatedly scanning an input signal within a user program ensures that consistent input information is always available. You define process image partition with STEP 7 when you assign addresses (which input/output addresses of the modules are listed in which process-image partition). The process image partition is updated by the user with SFCs.

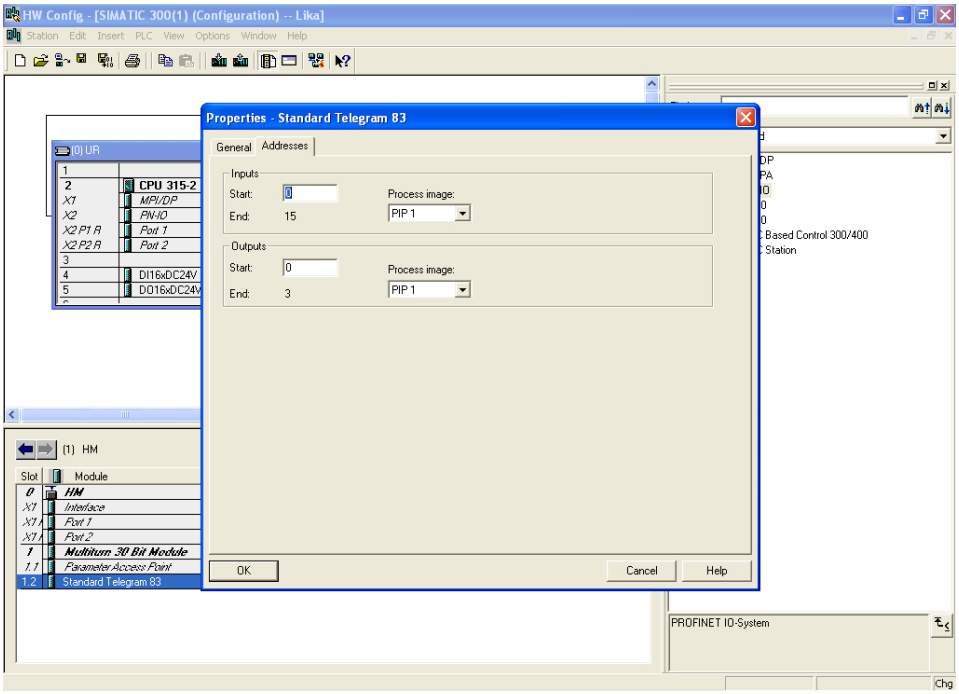


Figure 30 - Process Image Partition

The assigned process image partitions must be updated in the synchronous cycle interrupt OB61 at the call of the functions SFC126 "SYNC_PI" and SFC127 "SYNC_PO". The call of the function SFC126 "SYNC_PI" updates the process image partition input table; the call of the function SFC127 "SYNC_PO" updates the process image partition output table. So cyclic interrupt OB61 works with consistent image of the process signals. Refer also to page 126.

11.5.2 SFC126 "SYNC_PI"

This special function is used to update a process image partition input table in a synchronous cycle. Synchronous cycle interrupt OB61 calls the function SFC126 "SYNC_PI" to consistently and synchronously update the input data located in a process image partition.

11.5.3 SFC127 "SYNC_PO"

This special function update a process image partition output table in a synchronous cycle. Synchronous cycle interrupt OB61 calls the function SFC127 "SYNC_PO" to synchronously update the output data located in a process image partition and consistently transmit them to the encoder.

11.6 Domain Management

WARNING
The configuration of the Sync domain requires both in-depth skills and specific expertise in SIMATIC STEP 7 programming environment. For detailed information please consult the STEP 7 Programmer's handbook and documentation.

In the case of PROFINET IO with IRT, a sync master transmits a synchronization message to which all sync slaves synchronize themselves. The synchronization mechanisms will be controlled by ERTEC (Enhanced Real-Time Ethernet Controller) of the integrated PROFINET interface. This ensures a synchronization accuracy of less than one microsecond. The synchronization of all IRT-capable PROFINET devices on a common time base is the prerequisite for the scheduled communication control and the bandwidth reservation.
You assign the device roles sync master and sync slave by configuring the PROFINET devices in STEP 7, as described below. The role of a sync master can be assigned both to an IO controller as well as a switched configured on an IO device, provided these support the "sync master" function.
The sync master and the sync slaves together form the sync domain. A sync domain has exactly one active sync master - for the runtime.

It is mandatory that all PROFINET devices that are to be synchronized via PROFINET IO with IRT must belong to a sync domain. The sync domain consists of precisely one sync master and at least one sync slave. IO controllers and switches can hold the role of a sync master or sync slave. Other IO devices support only the role as sync slave. An IRT Class 3 Device must be member of a sync domain too. The properties of the sync domain must be set in the dialog boxes described in this section, especially the cycle time of the send clock. Only those values may be entered which are common to both the Master and all devices on the bus.

To assign the IO device to a sync domain select and right-click the **Profinet-IO system** in the **HW Config** window. Press the **PROFINET IO Domain Management...** command in the shortcut menu.

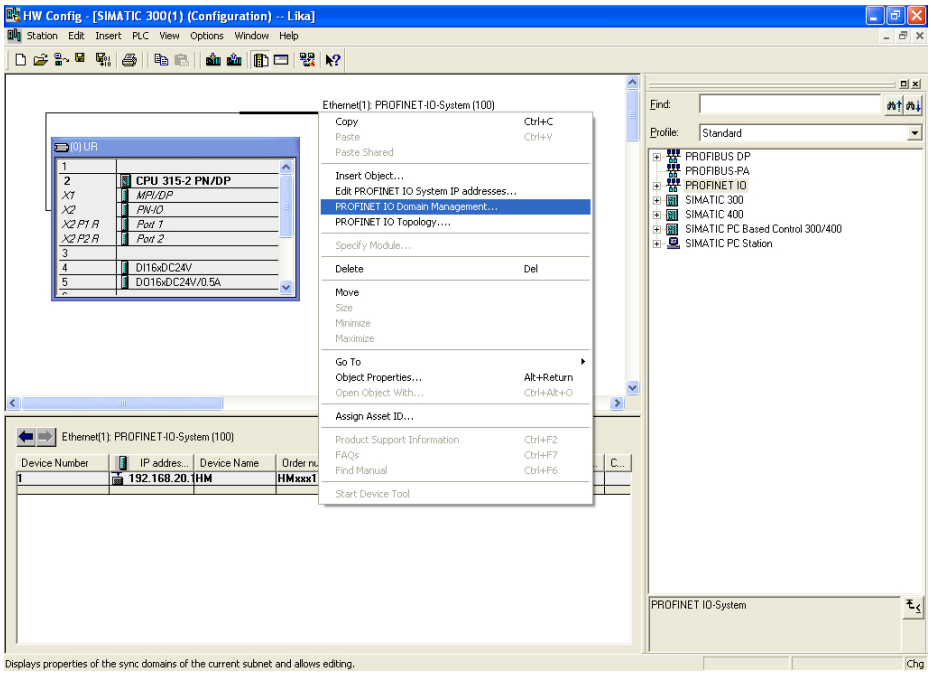


Figure 31 - Assigning the IO device to a sync domain

The **Domain Management** dialog box will appear on the screen.
Using the **Domain management** dialog box you can check the settings of the PROFINET IO system.

The name of the sync domain is automatically assigned by STEP 7 with "syncdomain-default" when the first sync domain is configured. You find it under the **Sync domain** item in the **Sync domain** group box. Alternatively you can create further sync domains.

The **Send clock time (ms)** is the period between two successive intervals for IRT or RT communication. The send clock is the shortest possible transmit interval for exchanging data. You have the option to establish the send clock for your sync domain in order to achieve an optimum coordination of the transmission bandwidth to the data volume. Depending on the PROFINET devices of the respective PROFINET IO systems, STEP 7 calculates the possible values that can be set. Select one of the default send clock time values in the drop-down list.

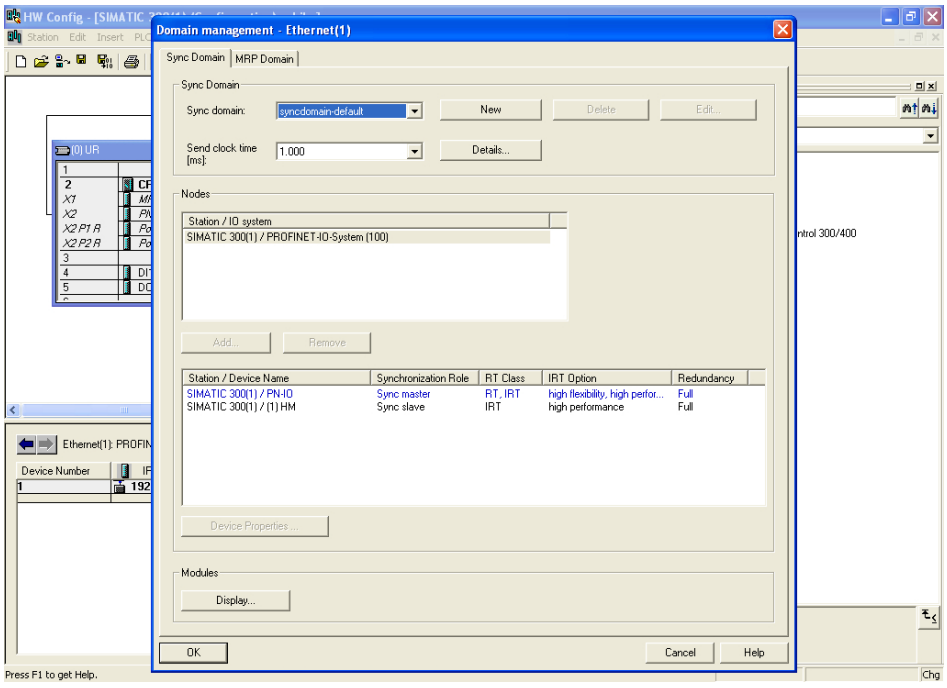


Figure 32 - Domain management dialog box

You have the option of establishing the reserved transmission bandwidth for IRT data as a proportion of the maximum reservable transmission bandwidth for cyclic user data communication. The proportion is specified in percent (%).
In doing so, the maximum transmission bandwidth set by the system for cyclic data may not be exceeded by the reserved transmission bandwidth for IRT plus the transmission bandwidth that is required for the cyclic data in the free transmission bandwidth (RT communication).

Press the **Details...** button in the **Domain Management** dialog box. The **Details – Sync domain** dialog box will appear on the screen. In the **Upper limit for IRT** drop-down list available in the **Cyclic data** group box, select one of the default values (in %). STEP 7 offers the following selections: 0, 10, ... 100.

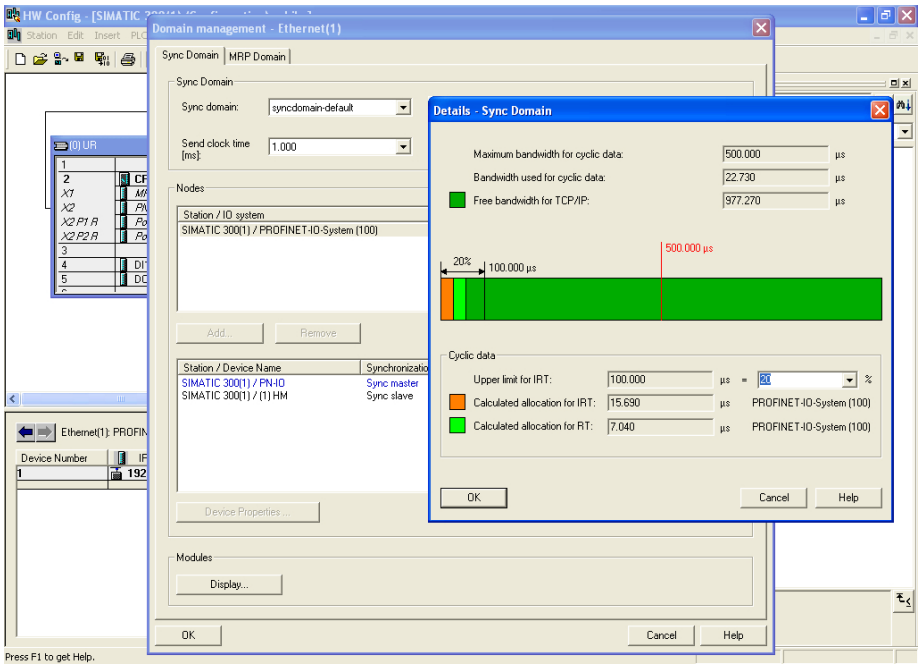


Figure 33 - Sync domain details dialog box

The PROFINET IO system of the sync domain is established now on a send clock.
Save the settings and close both the **Details – Sync Domain** and the **Domain management** dialog boxes pressing the **OK** button.
Finally save, compile and then download the configuration to the controller.

11.7 Topology Editor

Using the **Topology Editor** you can topologically configure your PROFINET IO system. The topology editor has a series of functions for setting, interconnecting and diagnosing the properties of all PROFINET devices including their ports. So it supports you in the following tasks:

- obtaining topology information about all ports of the PROFINET devices in the project;
- configuring the set topology on PROFINET by interconnecting the interfaces and ports by simply dragging and dropping and establishing properties.

Enter the **Topology Editor** dialog box to edit the topology of your Profinet-IO System. To do so select and right-click the **Profinet-IO system** in the **HW Config** window and then press the **PROFINET IO Topology...** command in the shortcut menu.

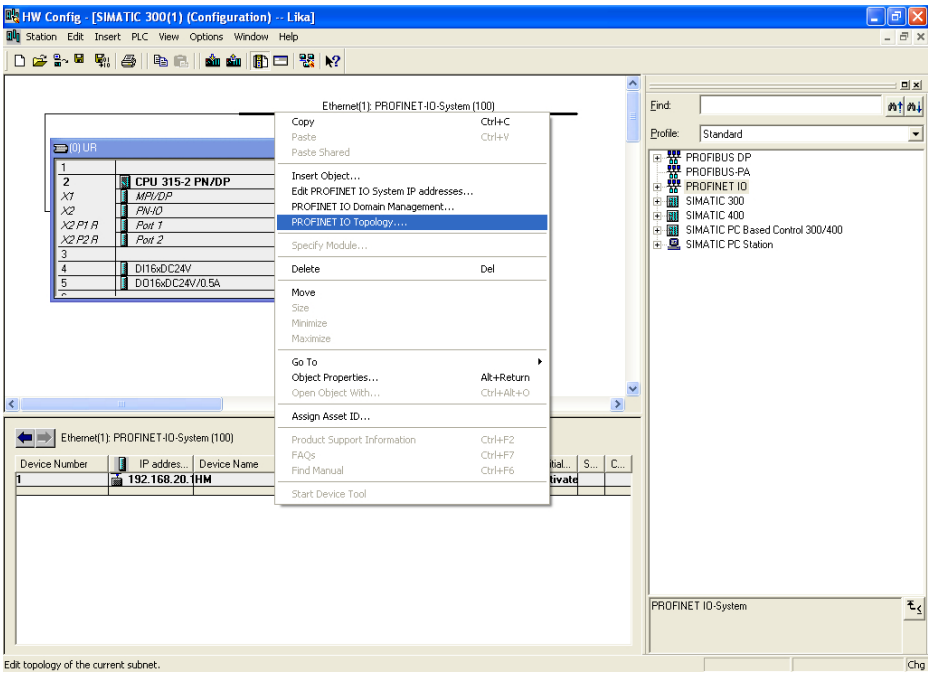


Figure 34 - Entering the Topology Editor

The **Topology Editor** dialog box will appear on the screen.
As the PROFINET IO system has already been set up and the PROFINET devices connected (to connect the port of the IO controller and the port of the IO device see on page 127), they do not need to be first of all interconnected to a topology within the project. So in the **Table view** tabbed page you will see the configured PROFINET devices and ports currently connected in the system.
The **Interconnection table** in the left area of the page lists all the configured PROFINET devices with their ports. The **Selection range** group box in the top right section lists all the PROFINET devices that are available for the topological interconnection.

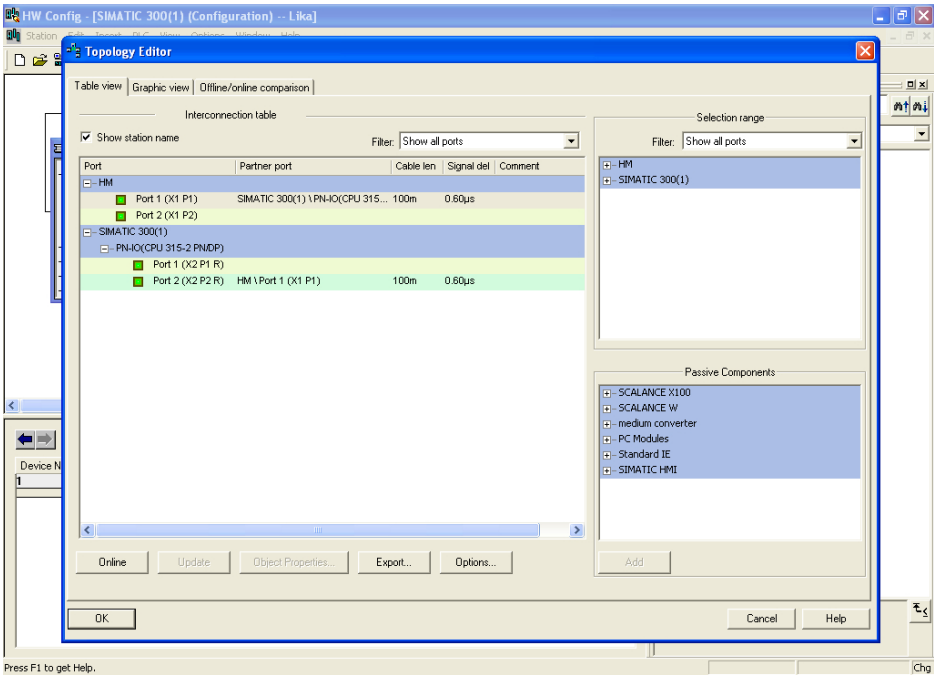


Figure 35 - Topology Editor: Table view

The **Offline/online comparison** tabbed page lists the configured PROFINET devices with their interfaces and ports and the associated neighbour ports in the left box **Configured topology (offline)**. The right box **Detected topology (online)** shows the online view of the PROFINET IO system with its interconnection for the runtime. Press the **START** button above in the box to import the PROFINET devices of your project.

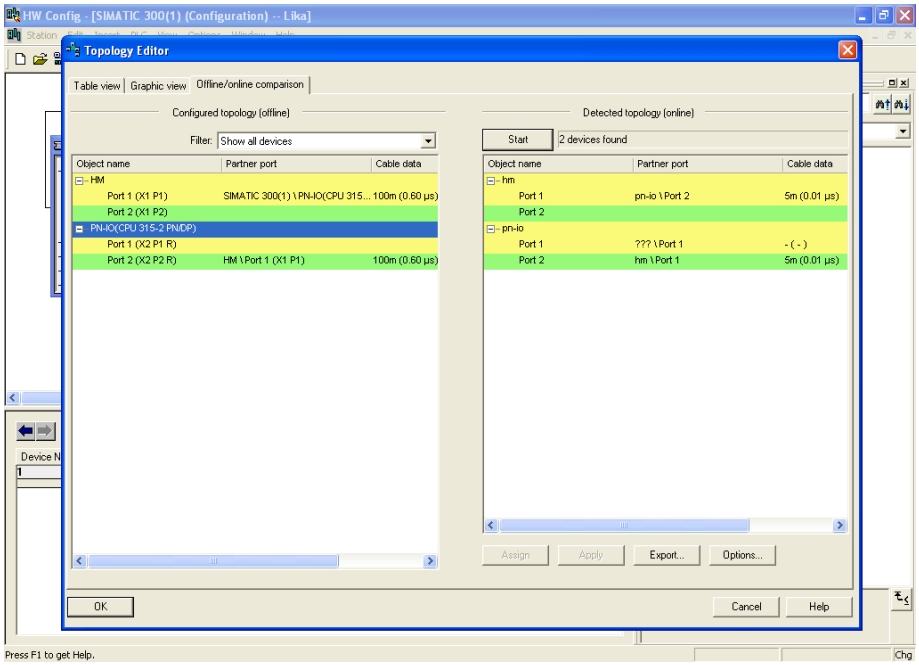


Figure 36 - Topology Editor: Offline/Online comparison

The **Graphic view** tabbed page displays the PROFINET devices in the project and their interconnections.
In the **Miniature View** in the top right section, you can use the slider to select the section of the PROFINET IO system and its enlargement factor. To change the section of the PROFINET IO system, use the mouse to drag the frame across the desired area that you wish to see in detail.

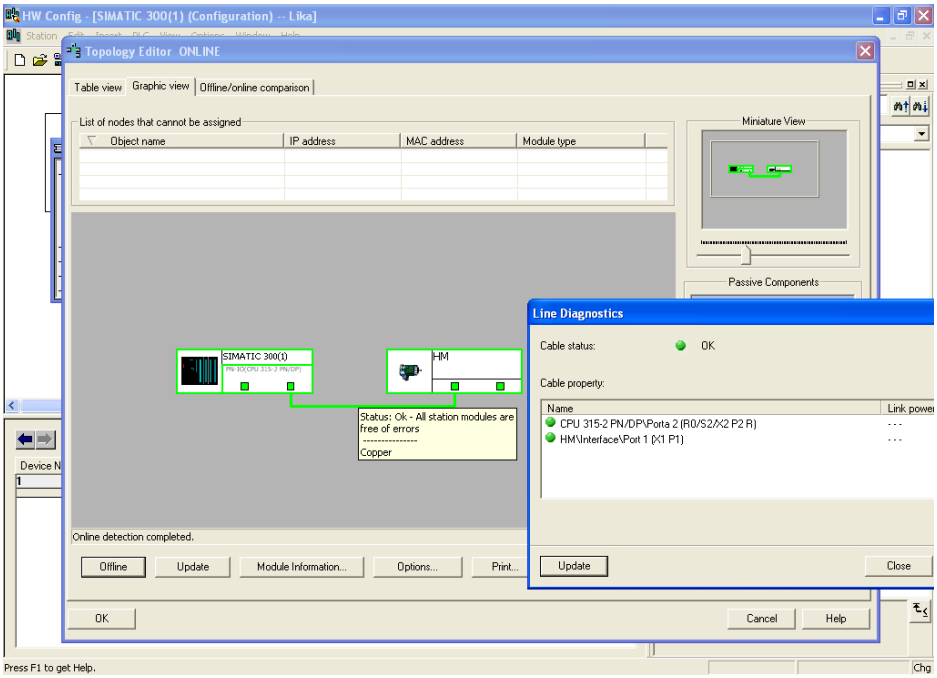


Figure 37 - Topology Editor: Graphic view

Save the settings and close the **Topology Editor** dialog page by pressing the **OK** button.

11.8 Message monitoring

Below is an example of traffic between the IO controller and the IO device in IRT mode.

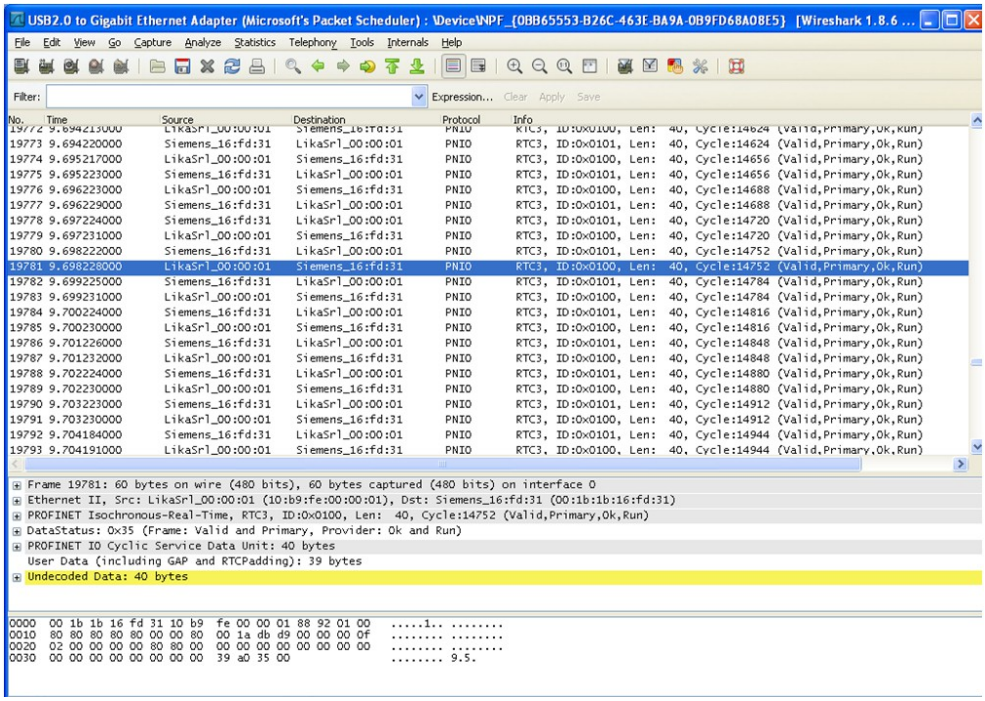


Figure 38 - Message monitoring

12 Encoder replacement using LLDP

LLDP (Link Layer Discovery Protocol) is a Layer 2 protocol that is used to detect the closest neighbours in the network. It enables a device to send information about itself and to save information received from neighbouring devices, i.e. it provides the option of communicating data between neighbouring devices (e.g. device name, port, MAC address). This information allows a network management system to determine the network topology. The protocol is formally referred to by the IEEE as *Station and Media Access Control Connectivity Discovery* specified in standards document IEEE 802.1AB.
Among the main uses, LLDP allows to replace a device of the Profinet network.
The partner ports before and behind the replaced device save the relevant information so that no additional configuration is necessary. The flag **Support device replacement without exchangeable medium** must be activated in the Controller.
To activate / deactivate the function double click the **PN-IO** slot X2 to open the PN-IO properties dialog box. The **Properties – PN-IO** property sheet will appear. Enter the **General** tabbed page to find the **Support device replacement without exchangeable medium** check box. Please note that the **Support device replacement without exchangeable medium** function is activated by default in the IO controller.

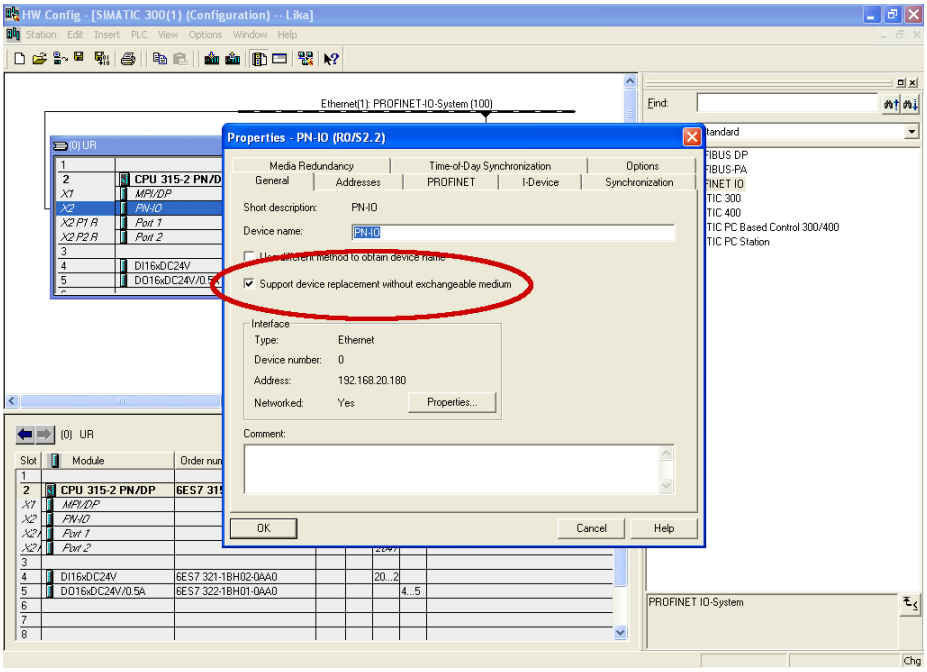
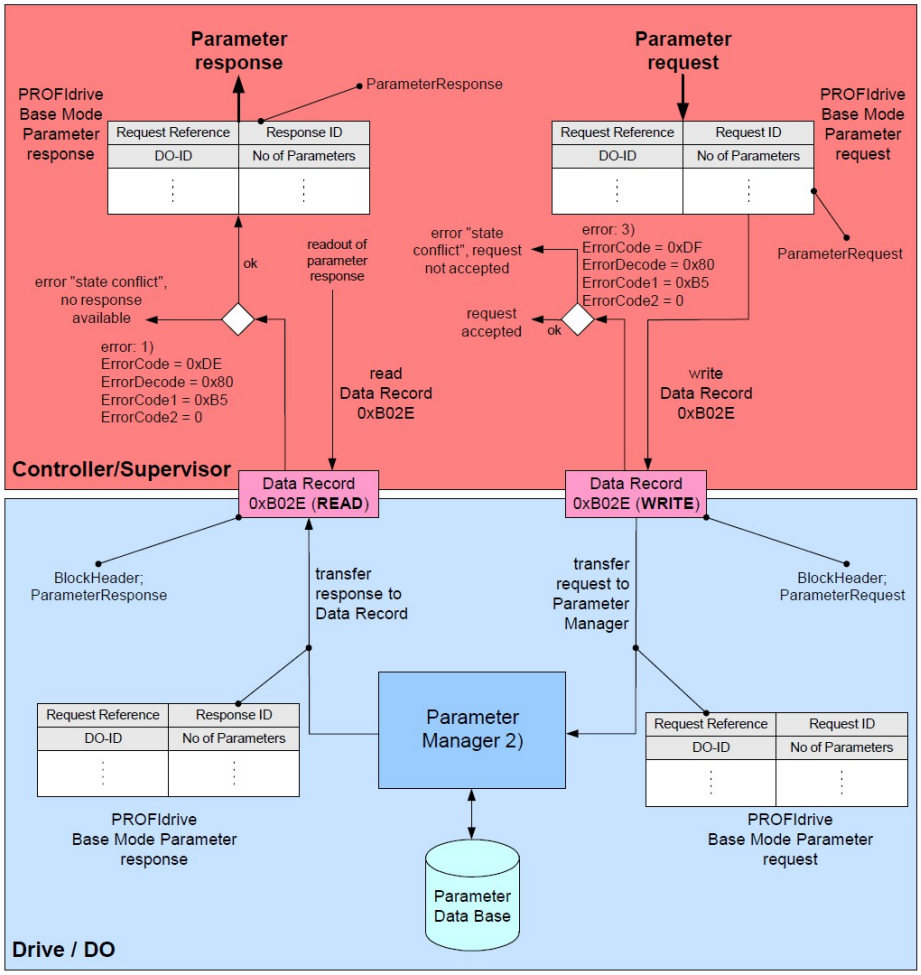


Figure 39 - Link Layer Discovery Protocol (LLDP)

NOTE

When you replace a device, make sure that the PROFINET cable is then inserted into the correct port as it is configured in STEP 7. Otherwise, the system will not run.

13 Read & write in acyclic comm



1) Error because the parameter manager is busy but not finished with the processing yet, or the parameter manager is idle

2) Processing of only one parameter requests per connection. Multiple connections cause multiple state machines for the processing each for every connection

3) Also error 0xB0 may be used if there is no PAP available and error 0xB7 if there is an error in the request header

Figure 40 - Base mode parameter request and response

13.1 Example: reading and writing a parameter (Preset Value)

13.1.1 System Function Block 52 (SFB52)

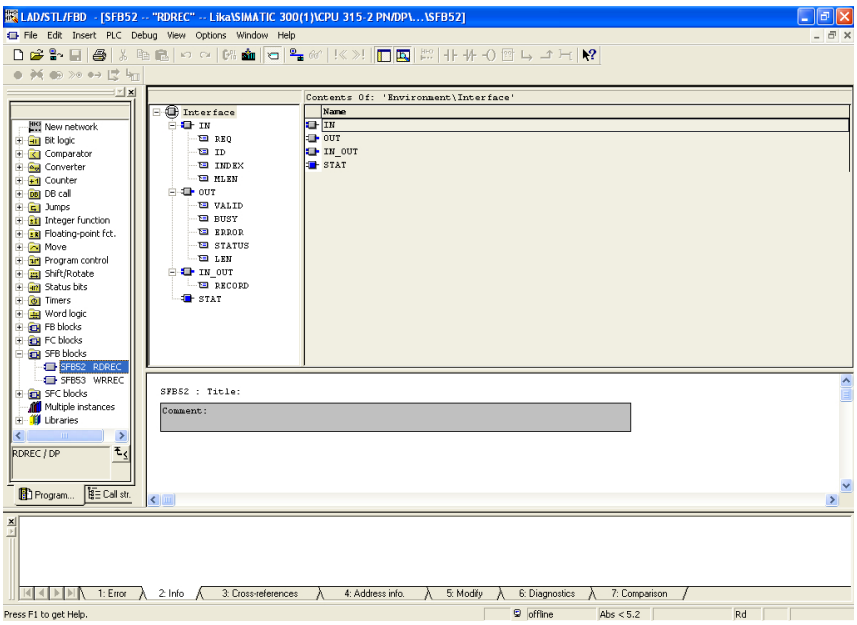
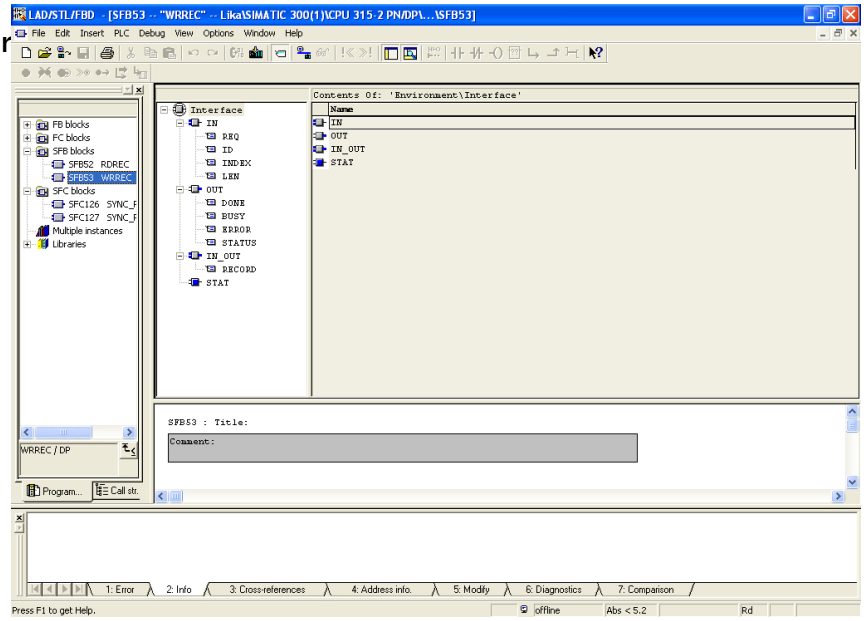


Figure 41 - SFB52

13.1.2 System Function Block 53 (SFB53)

Figur



13.1.3 Data Block 1 (DB1)

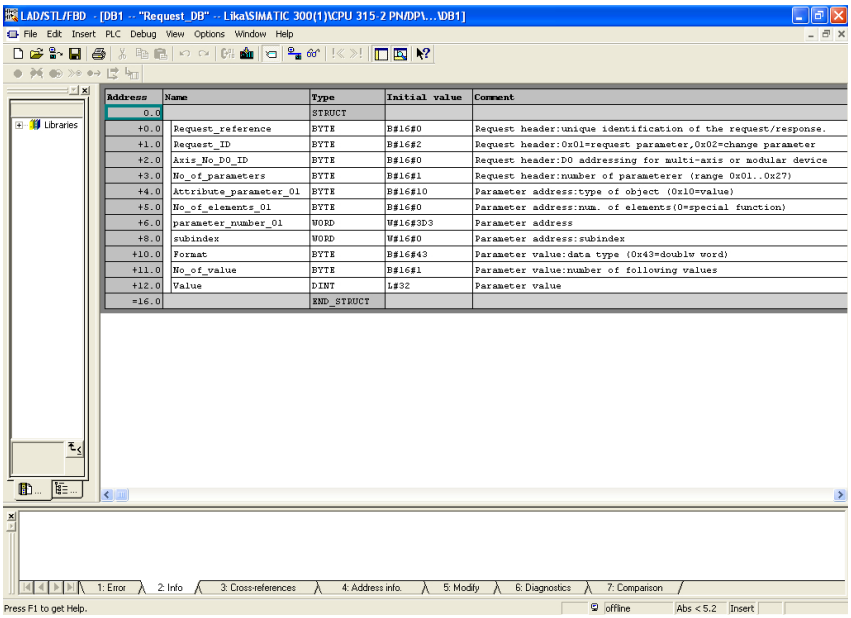


Figure 43 - DB1

13.1.4 Data Block 2 (DB2)

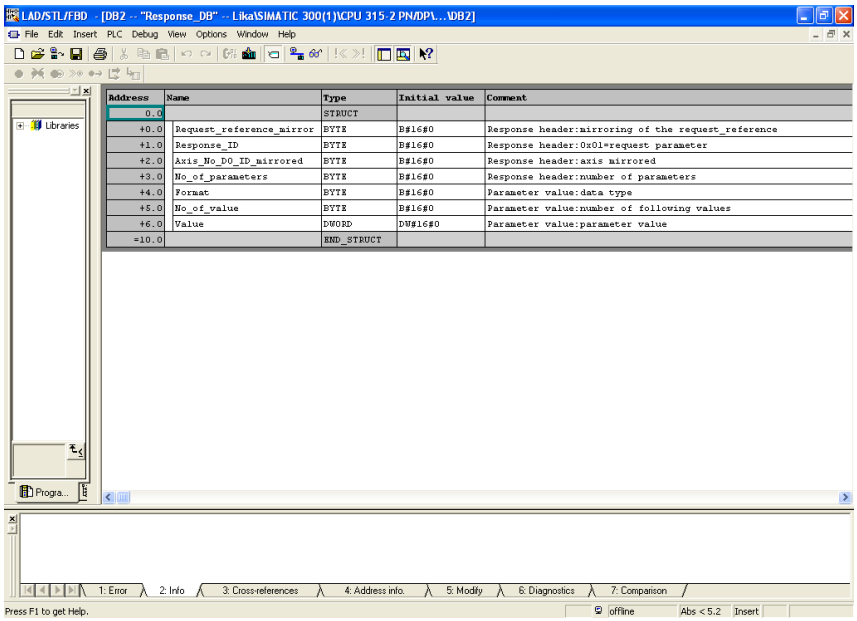


Figure 44 - DB2

13.1.5 Data Block 3 (DB3)

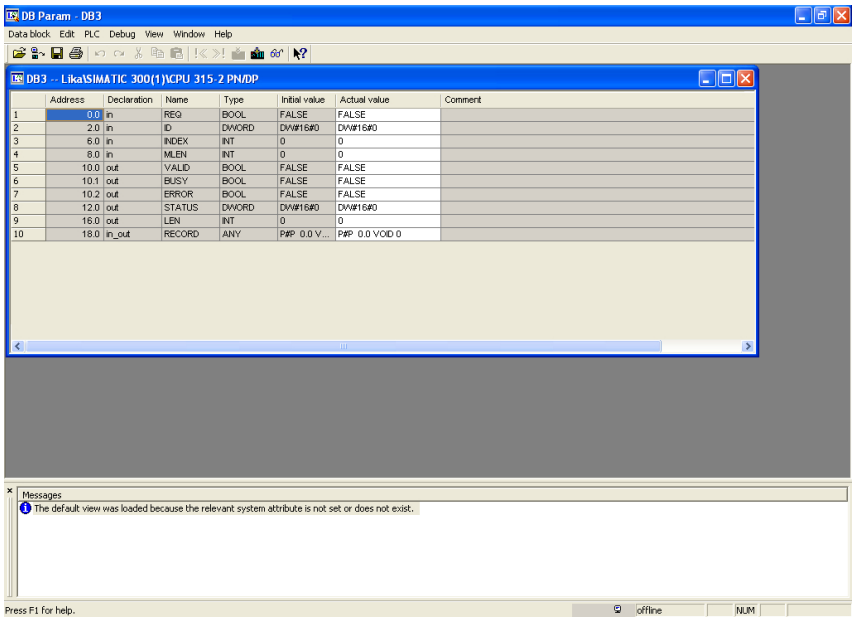


Figure 45 - DB3

13.1.6 Data Block 4 (DB4)

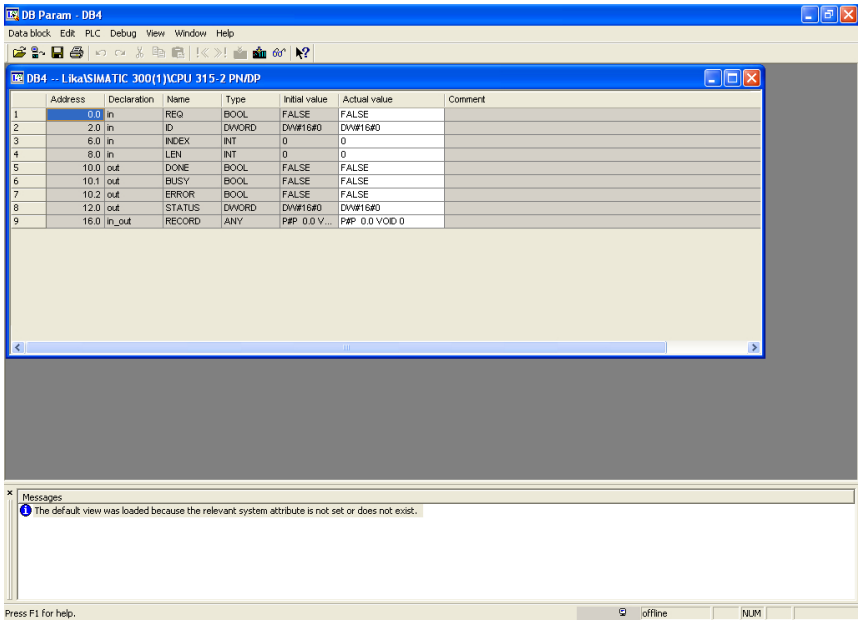


Figure 46 - DB4

13.1.7 Organization Block 1 (OB1)

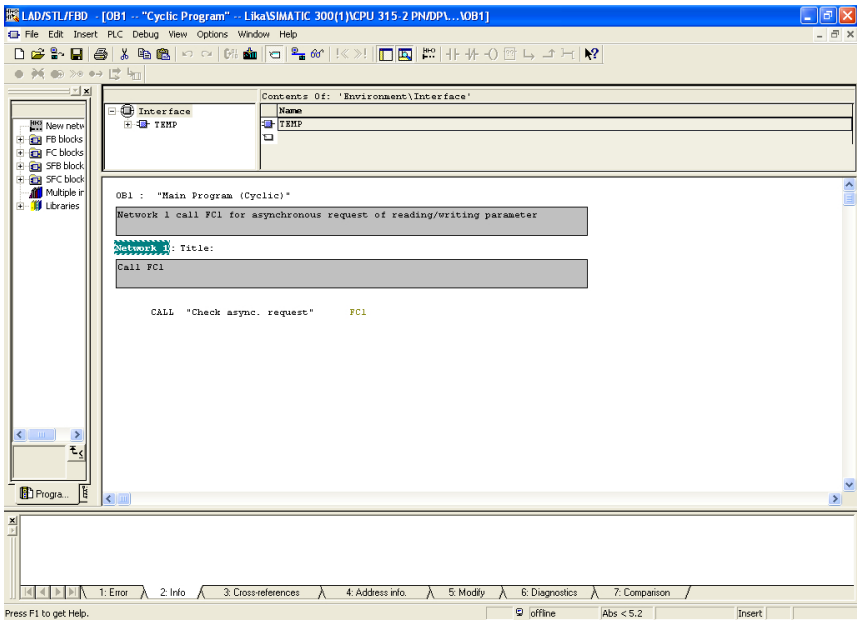
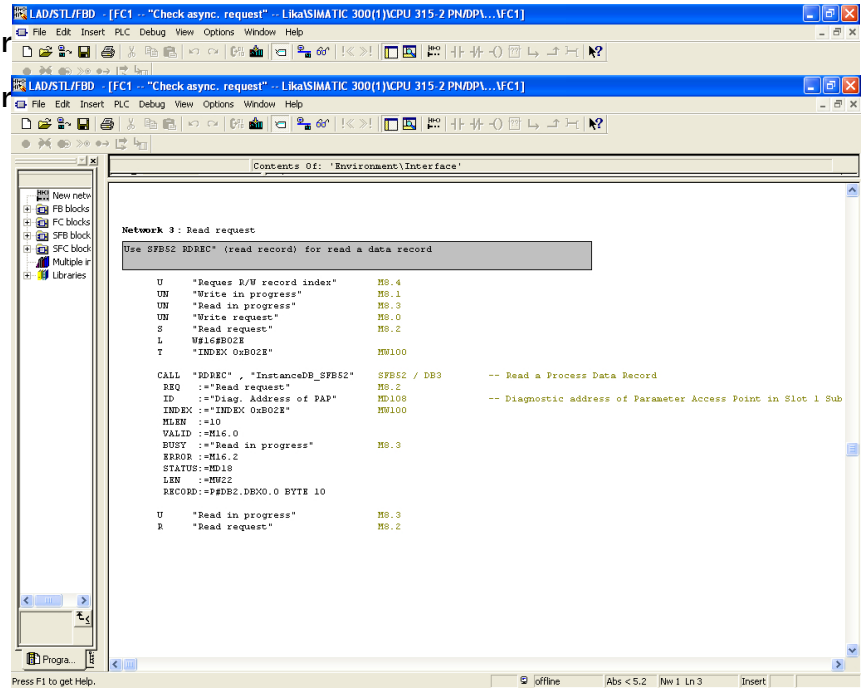


Figure 47 - OB1

13.1.8 Function 1 (FC1)



13.1.9 Function 2 (FC2)

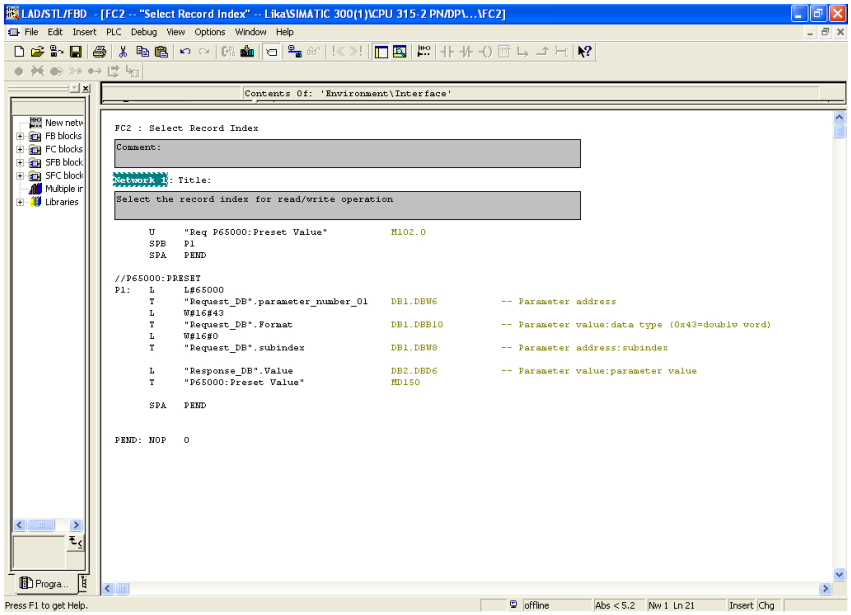
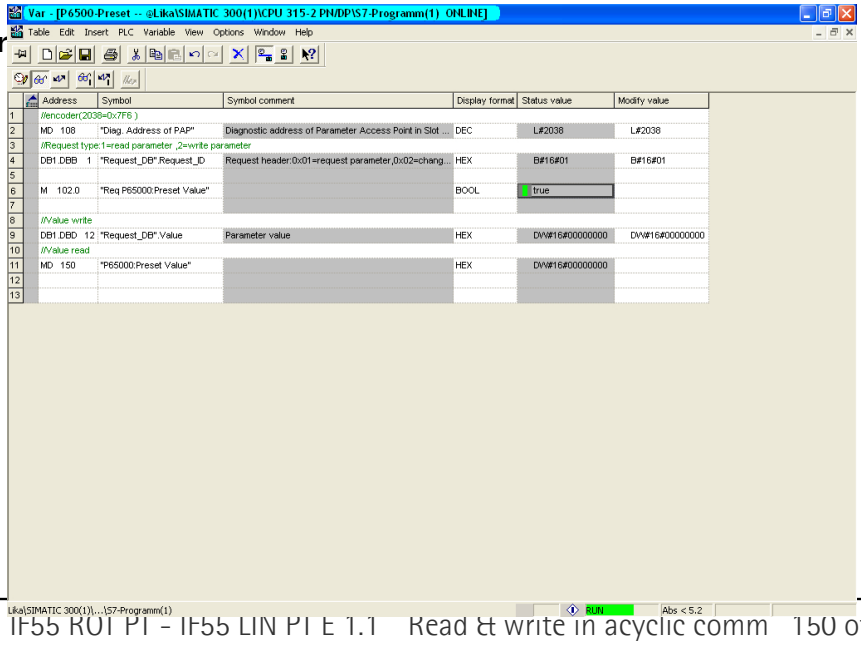


Figure 50 - FC2

13.1.10 Acyclic request of Preset

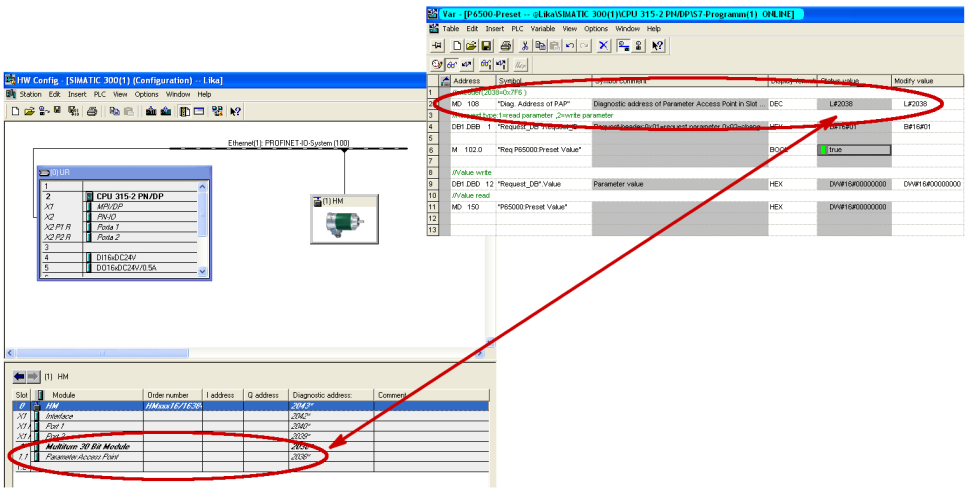
See **P65000 – Preset value** on page 83.

Figur



NOTE

Please always ascertain that **MD 108 Diag. Address of PAP** is the same as the **diagnostic address of Parameter Access Point in Slot 1.1**.



13.2 Monitoring a variable

Below is an example of variable monitor in case of Telegram 83 and IRT communication.

NOTE
Controller Sign-Of-Life is active.

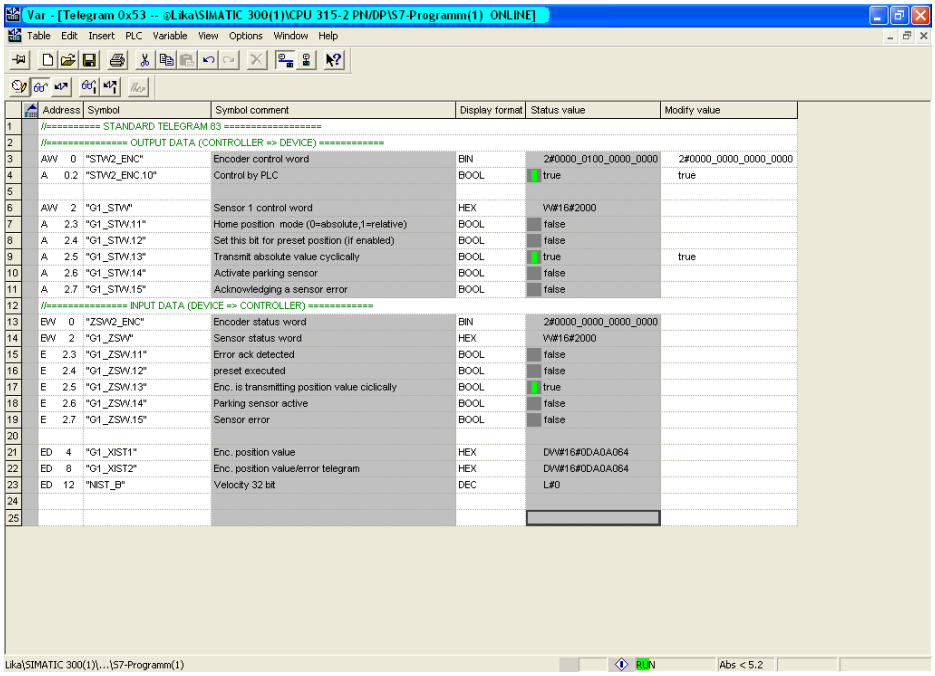


Figure 52 - Monitoring a variable

14 Encoder state machine

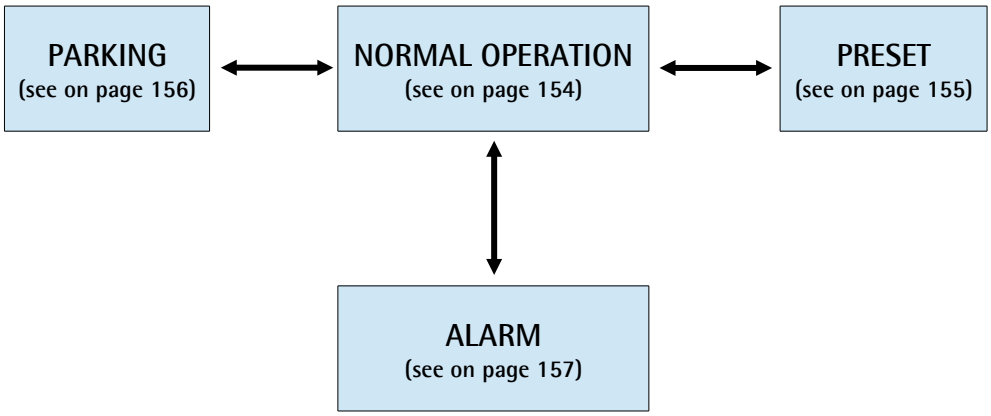
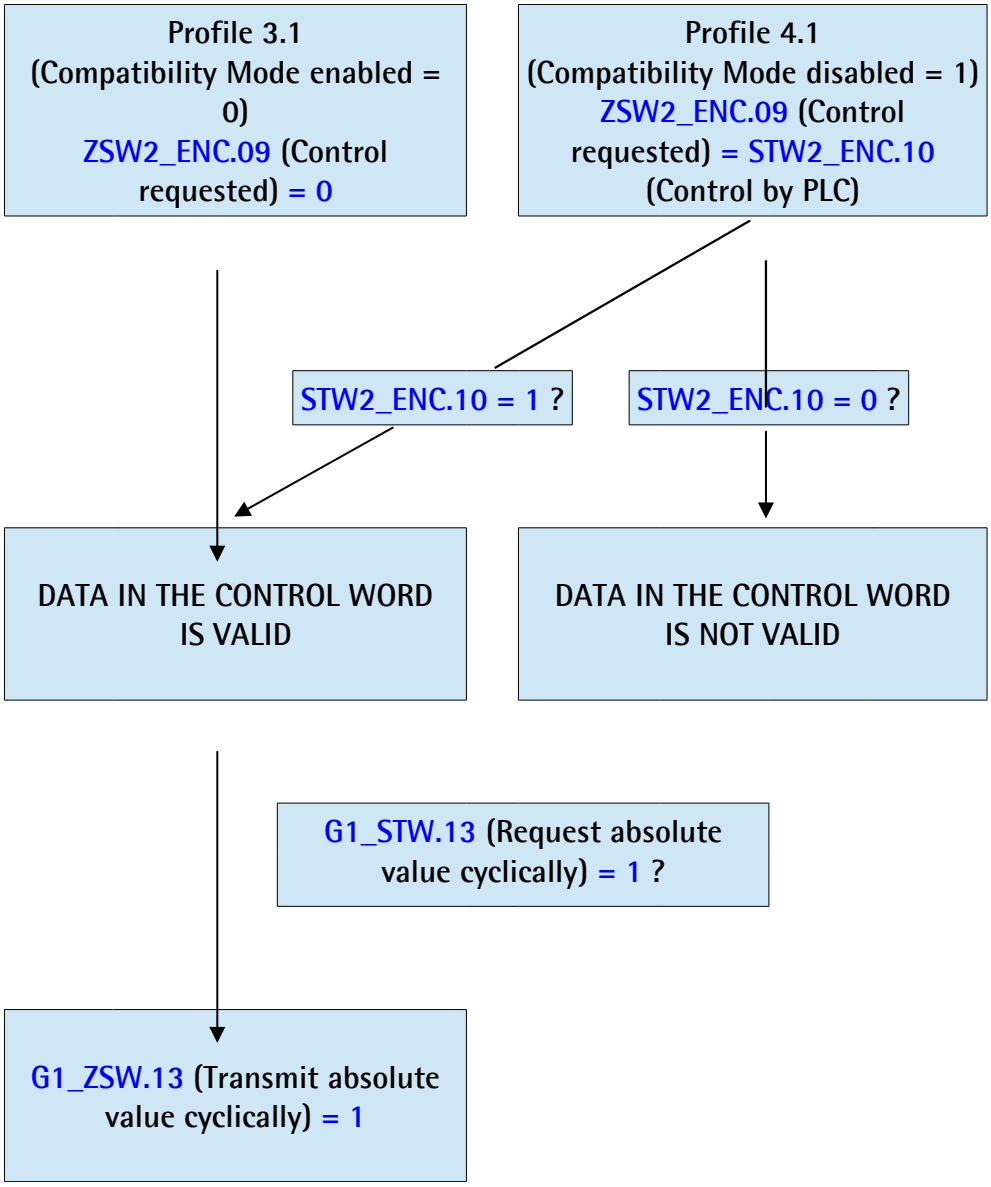
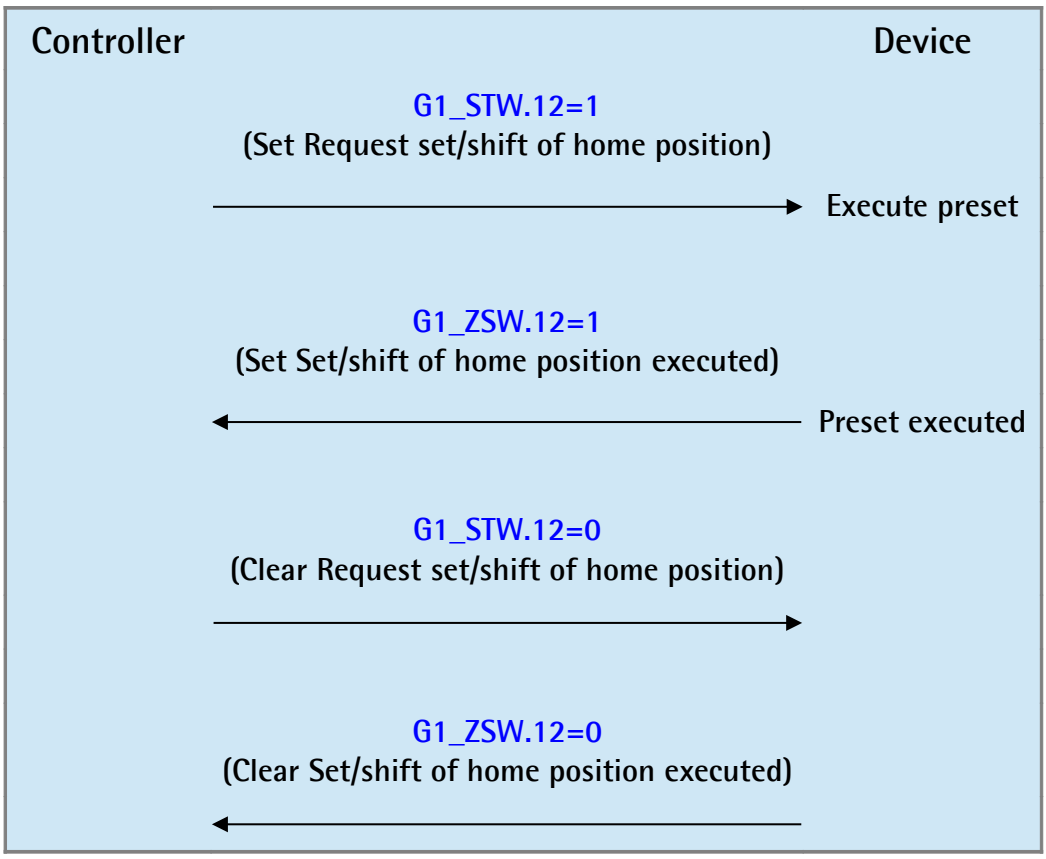


Figure 53 - Encoder state machine

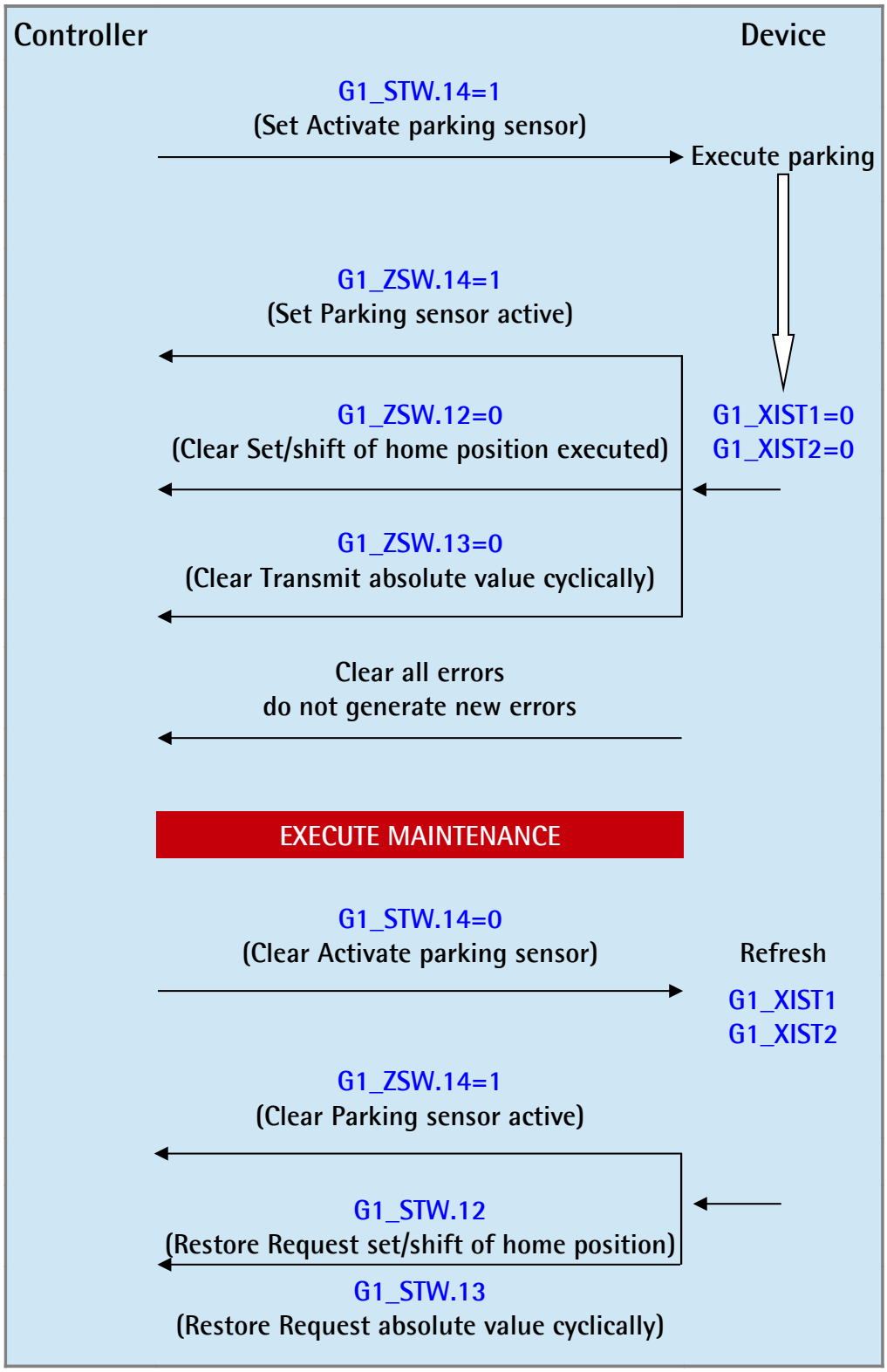
14.1 Normal operation diagram



14.2 Preset diagram

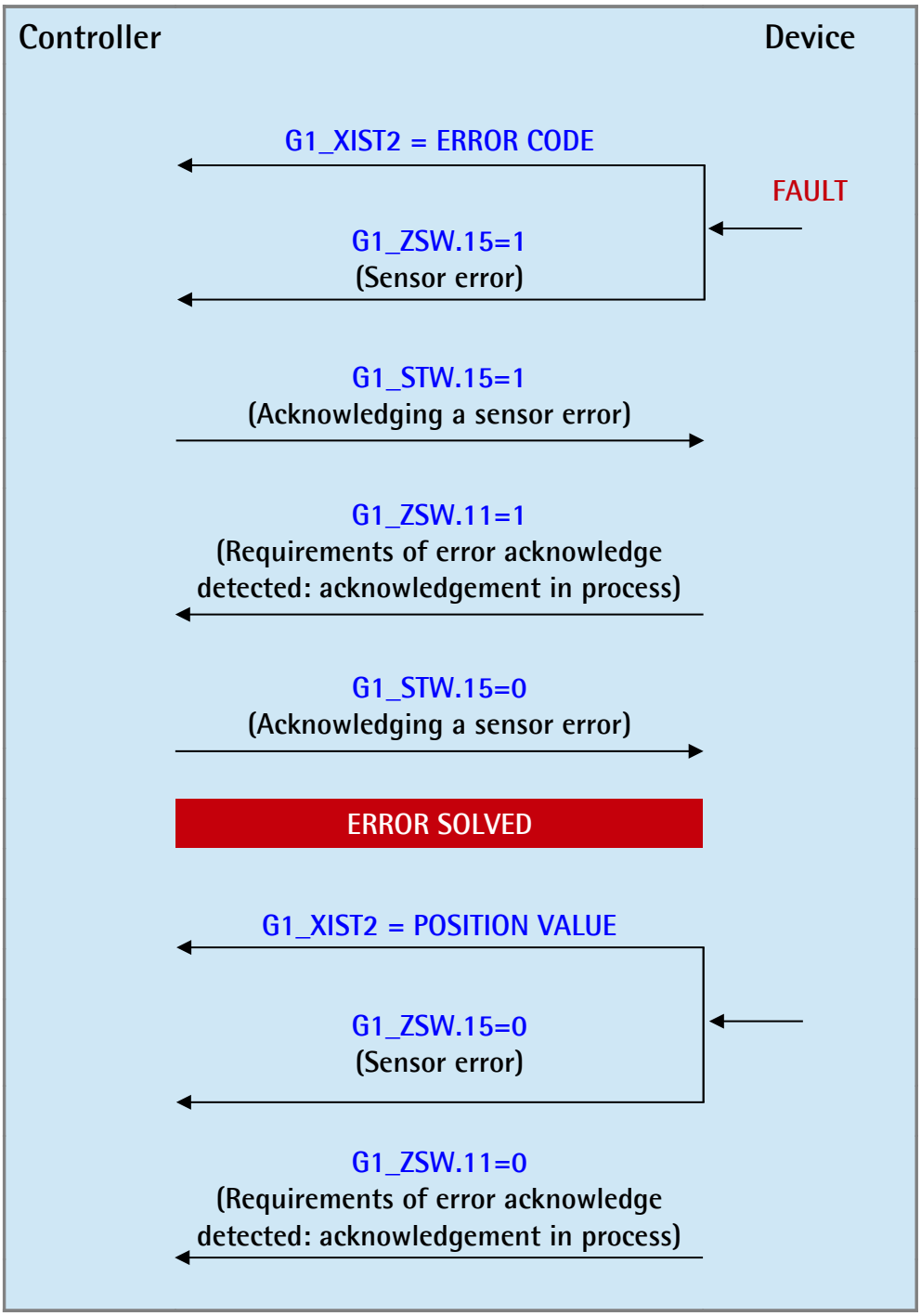


14.3 Parking sensor diagram

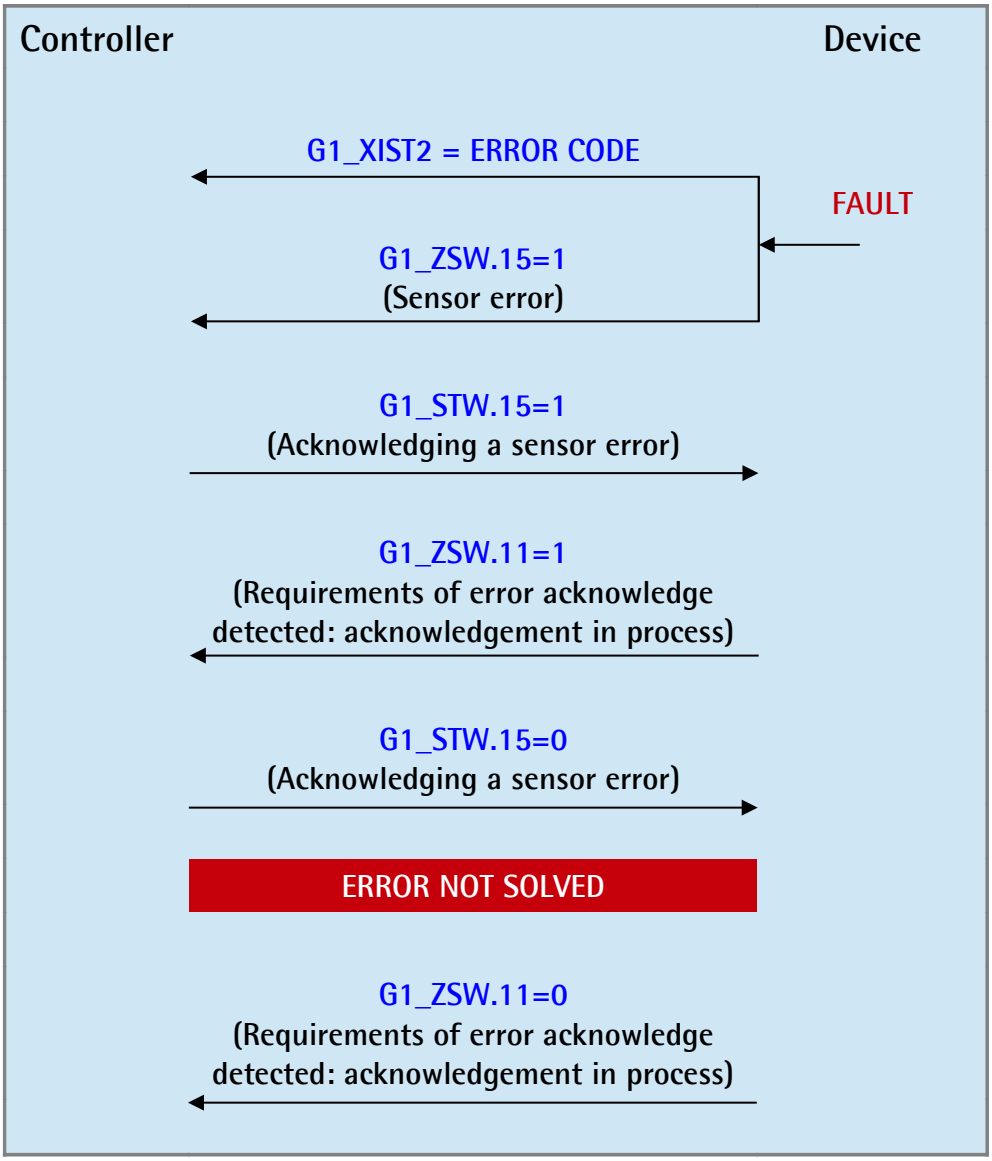


14.4 Error diagram

14.4.1 Acknowledgement of acknowledgeable sensor error



14.4.2 Acknowledgement of not acknowledgeable sensor error



15 Integrated web server

Profinet gateways from Lika Electronic integrate a web server. This web-based user interface is designed to offer helpful functions and deliver complete information on the device that can be accessed through the Internet.
In particular it allows:

- to display and check the currently set parameters;
- to set the network communication parameters;
- to set some parameters such as the preset and the code sequence;
- to upgrade the firmware;
- to monitor the encoder and access some advanced maintenance functions.

The web server can be accessed from any PC running a web browser. Since its only requirement is a HTTP connection between the web browser and the web server running on the device, it is perfectly fitted also for remote access scenarios.

Before opening the Profinet encoder web server please ascertain that the following requirements are fully satisfied:

- the encoder is connected to the network;
- the encoder has valid device name and IP address;
- the PC is connected to the network;
- a web browser (Internet Explorer, Mozilla Firefox, Google Chrome, Opera, ...) is installed in the PC or in the device used for connection.

NOTE
This web server has been tested and verified using the following web browsers:

- Internet Explorer IE11 version 11.1593.14393.0
- Mozilla Firefox version 55.0.3
- Google Chrome version 60.0.3112.113
- Opera version 47.0.2631.80

NOTE
Please note that the snapshot look may vary depending on the used web browser. The following snapshots have been taken from Internet Explorer.

To open the Profinet encoder web server proceed as follows:

1. type the IP address of the encoder you want to connect to (in the example: 192.168.20.195) in the address bar of your web browser and confirm by pressing **ENTER**;

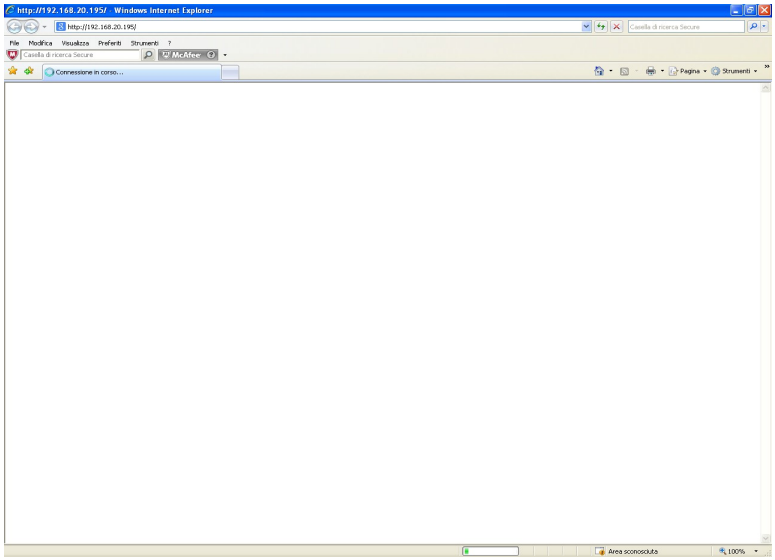


Figure 54 - Opening the web server

2. as soon as the connection is established, the web server Home page will appear on the screen;

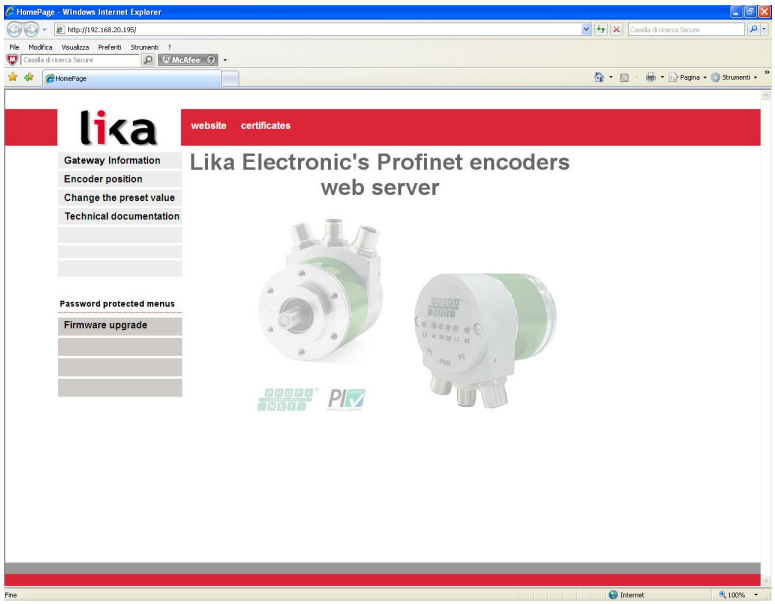


Figure 55 - Web server Home page

In the Home page some commands are available in the menu bar.
Press the **website** command to enter Lika's web site (www.lika.biz).
Press the **certificates** command to enter the Product certifications page on Lika's web site (http://www.lika.it/eng/certificazioni.php).

Furthermore some commands are available in the left navigation bar. The menu bar is divided into two sections: the commands in the above section allow to enter freely accessible pages; while the commands in the below section (under the label **Password protected menus**) allow to enter password protected pages.
These commands allow to enter specific pages where information and diagnostics on the connected encoder can be achieved.
They are shown in the following snapshots.

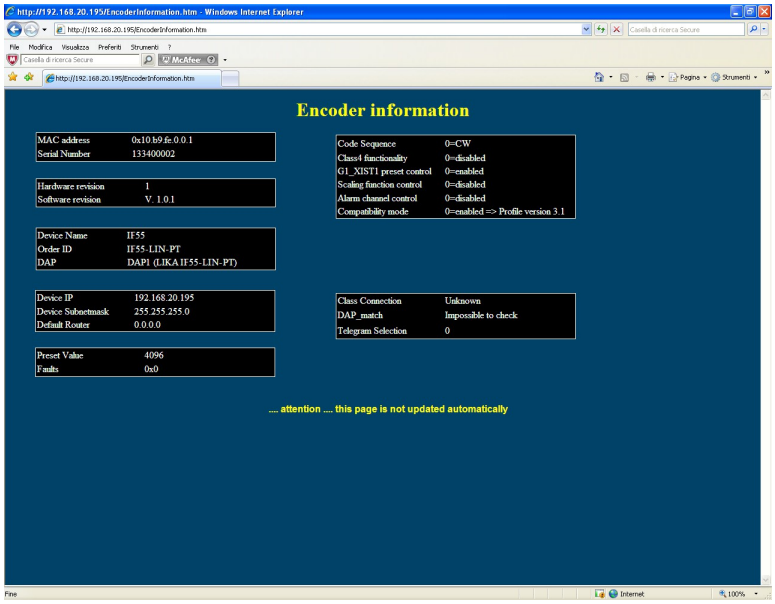


Figure 56 - Encoder Information page

Parameters listed in the **Encoder Information page** are sent by the controller to the encoder during initialization (for further information refer to the "9.4 Index 0xBF00 : user parameter data" section on page 86).

For further information on the encoder specific profile parameters refer to the sections "9.2 Index 0xB02E : supported PROFIdrive specific parameters" on page 80 and "9.3 Index 0xB02E : supported encoder specific parameters" on page 83.

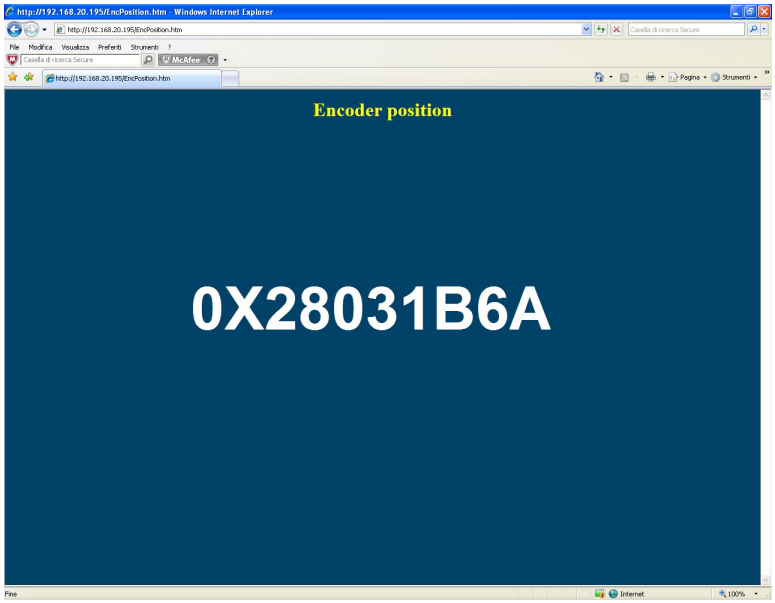


Figure 57 - Encoder position page

In the Encoder position page the current encoder position is shown.

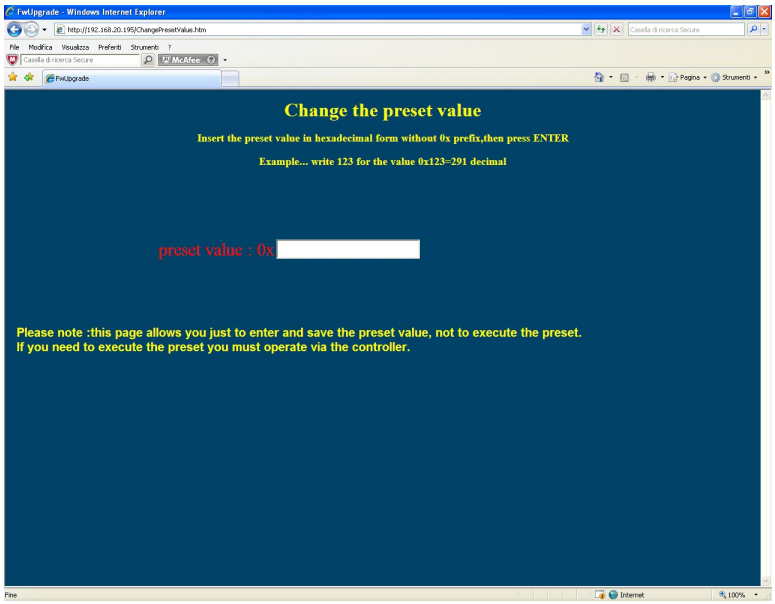


Figure 58 - Change the preset value page

For detailed information on setting the preset value please refer to the "15.2 Setting the preset value" section on page 171.

The **technical documentation** command allows to enter the web page (http://www.lika.it/eng/prodotti.php?id_cat=264&id_fam=276) in the corporate web site where specific technical information and documentation concerning the Profinet gateway can be found.

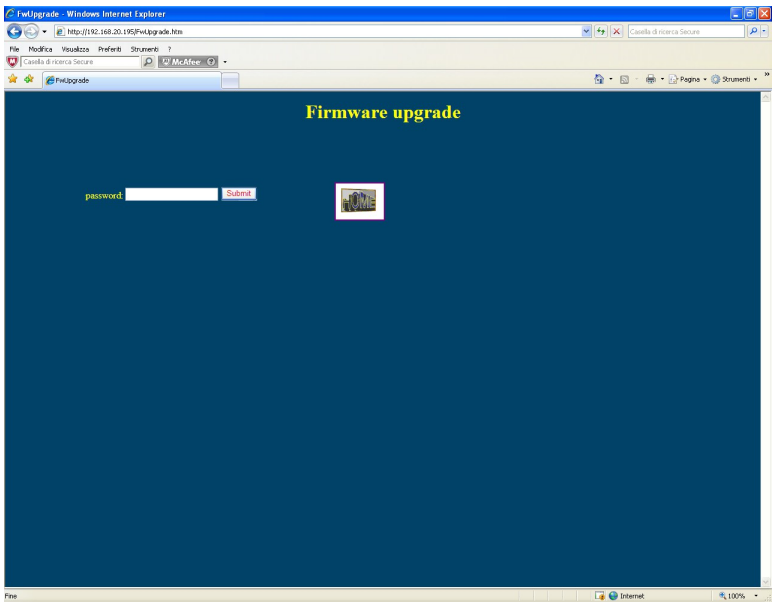


Figure 59 - Firmware upgrade page

This is a password protected page. For detailed information on the firmware upgrade please refer to the "15.1 Firmware upgrade" section on page 164.

15.1 Firmware upgrade

WARNING
Firmware upgrading process has to be accomplished by skilled and competent personnel. It is mandatory to perform the upgrade according to the instructions provided in this section.
Before installation always ascertain that the firmware program is compatible with the hardware and software of the device. Furthermore never turn off power during flash upgrade. In case of flash upgrade error, the program is lost irreversibly (there is not a bootloader) and the device must be sent back to Lika Electronic for restoring.

This operation allows to upgrade the unit firmware by downloading upgrading data to the flash memory.
Firmware is a software program which controls the functions and operation of a device; the firmware program, sometimes referred to as "user program", is stored in the flash memory integrated inside the unit. These encoders are designed so that the firmware can be easily updated by the user himself. This allows Lika Electronic to make new improved firmware programs available during the lifetime of the product.
Typical reasons for the release of new firmware programs are the necessity to make corrections, improve and even add new functionalities to the device.
The firmware upgrading program consists of a single file having .BIN extension. It is released by Lika Electronic Technical Assistance & After Sale Service.

If the latest firmware version is already installed in the unit, you do not need to proceed with any new firmware installation. Current firmware version can be verified in the Encoder Specific Profile Parameters page after connection to the web server.

NOTE
If you are not confident that you can perform the update successfully please contact Lika Electronic Technical Assistance & After Sale Service.

Before proceeding with the firmware upgrade please ascertain that the following requirements are fully satisfied:

- the encoder is connected to the network;
- the encoder has valid device name and IP address;
- the PC is connected both to the network and the IO controller;

- a web browser (Internet Explorer, Mozilla Firefox, Google Chrome, Opera, ...) is installed in the PC or device used for connection;
- you have the SW_PN.EXE executable file;
- you have the .BIN file for firmware upgrade.

To upgrade the firmware program please proceed as follows:

1. open the Profinet encoder web server by typing the IP address of the encoder you want to connect to (in the example: 192.168.20.195) in the address bar of your web browser and confirm by pressing **ENTER**;

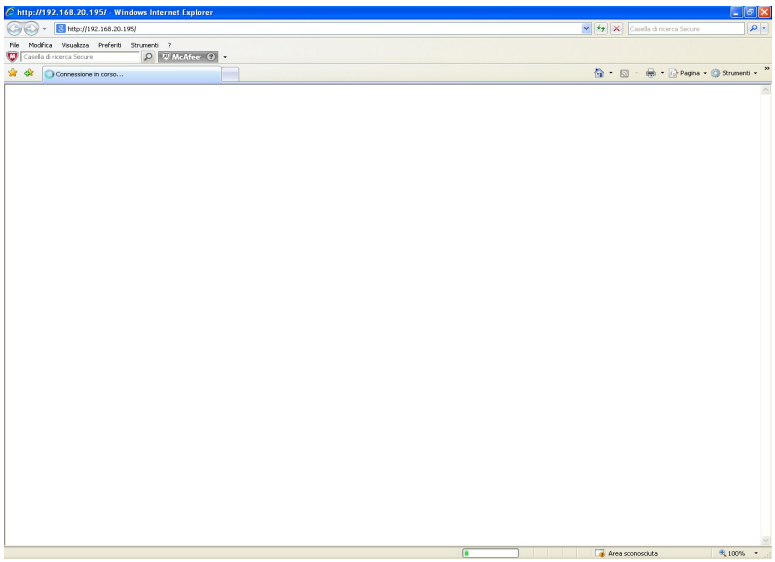


Figure 60 - Opening the web server

2. in the Home page press the **Firmware upgrade** command to enter the **Firmware upgrade** page;

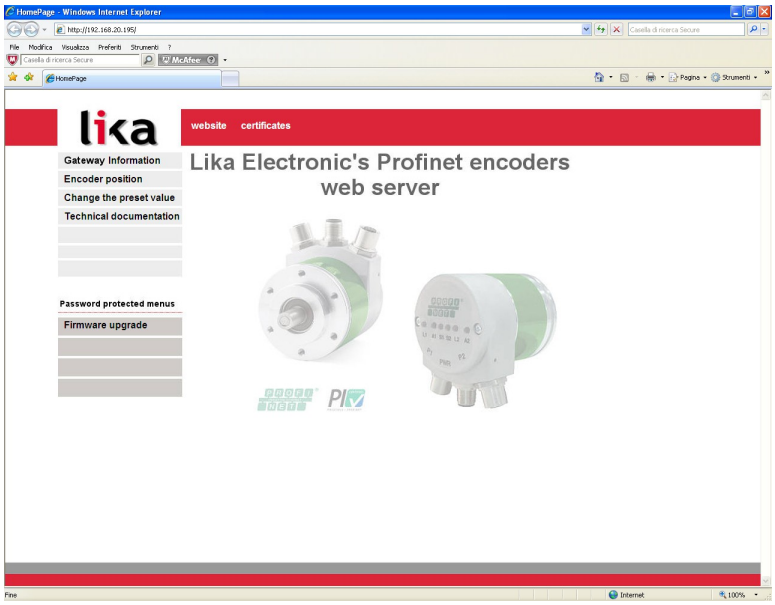


Figure 61 - Web server Home page

3. before entering the page you are requested to confirm the operation; press **OK** to continue;

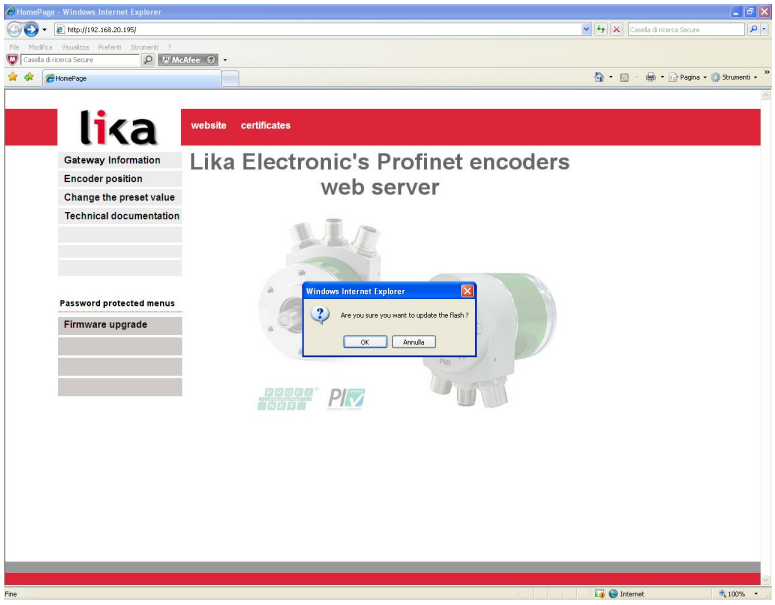


Figure 62 - Confirming the access to the Firmware upgrade page

4. in the **Password** text box type the password **lika** and then press the **Submit** button;

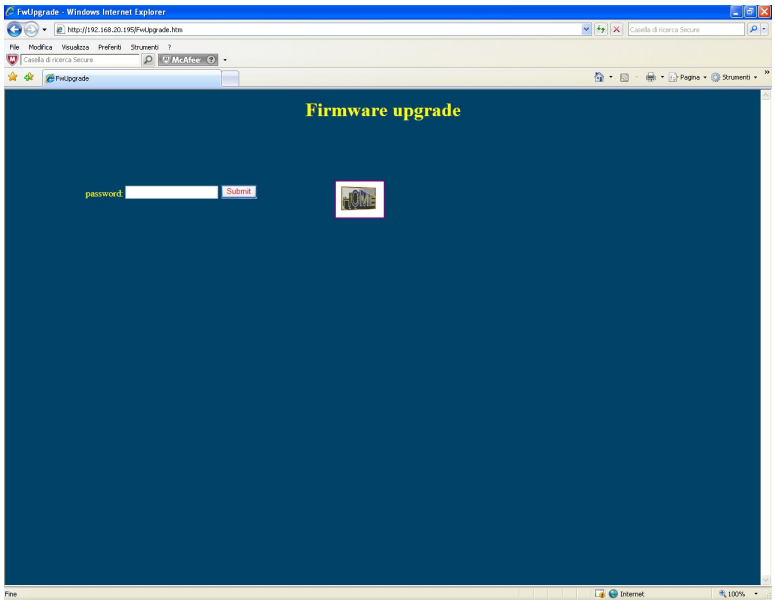


Figure 63 - Firmware upgrade page

5. a message will appear on the screen while the background colour of the page will be changed to red; now the web server is stopped and the encoder is ready to accept the firmware program;

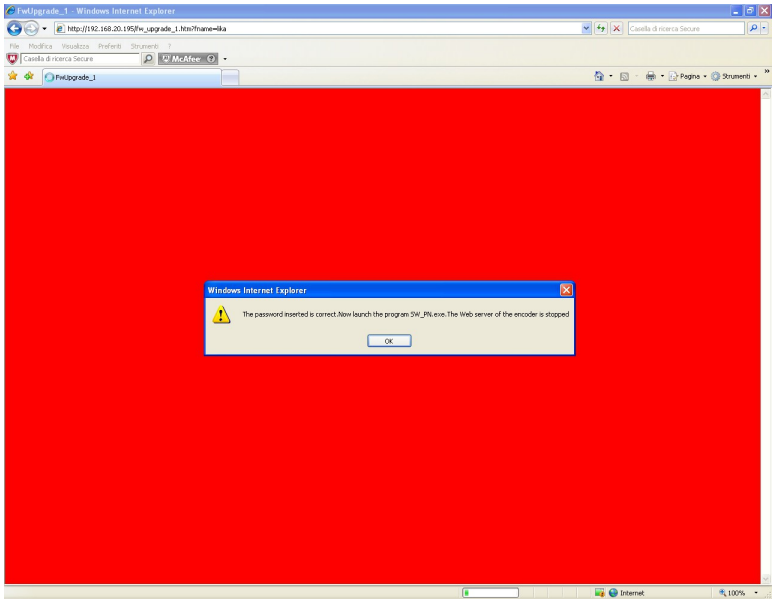


Figure 64 - Web server stopped

6. as soon as you press the **OK** button the following message will appear on the page: now you must launch the SW_PN.EXE executable file to continue with the procedure;

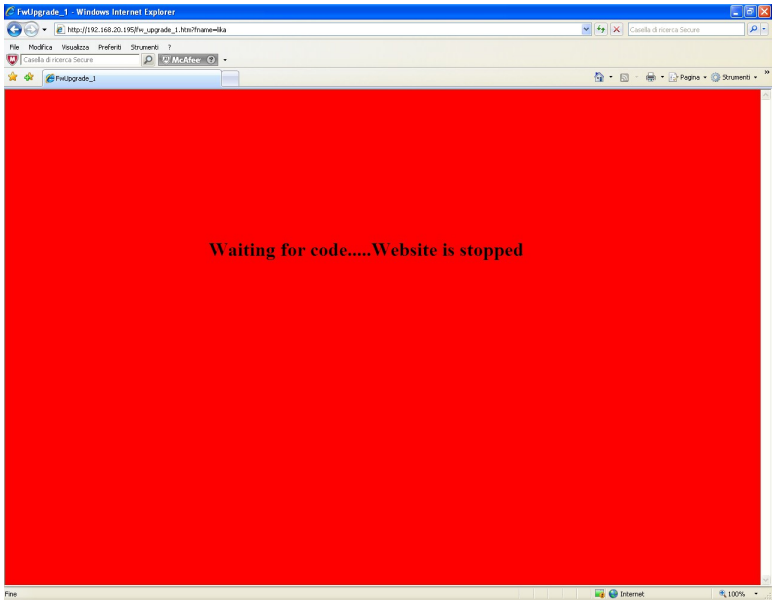


Figure 65 - Web server stopped

7. launch the SW_PN.EXE executable file provided with the technical documentation;
8. in the page that appears press the **SELECT FILE** button; once you press the button the **Open** dialogue box appears on the screen: open the folder where the firmware upgrading .BIN file released by Lika Electronic is located, select the file and confirm;

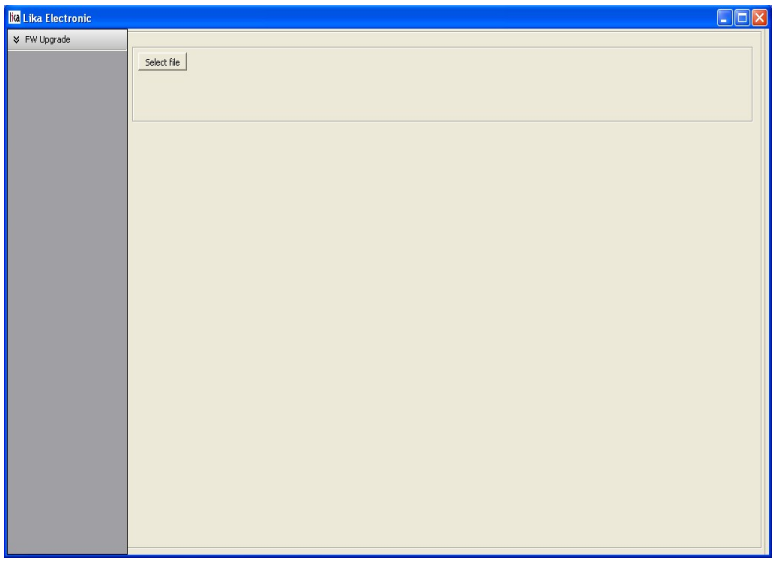


Figure 66 - Firmware upgrade executable file

9. in the **Remote host** text box type the IP address of the encoder you need to update; leave the **Remote port** box unchanged;
10. to start the upgrade press the **Upgrade FW** button;

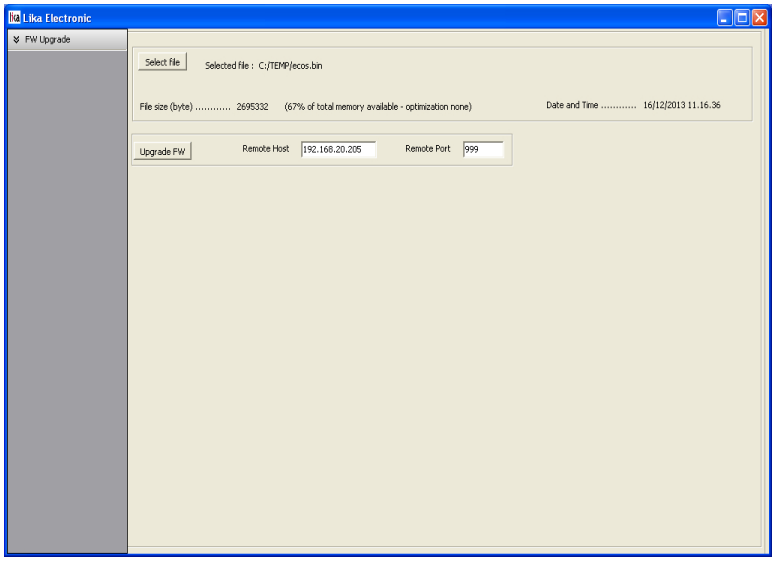


Figure 67 - Starting the firmware upgrade operation

WARNING

Before installation always ascertain that the firmware program is compatible with the hardware and software of the device.
Never turn off power during flash upgrade.
In case of flash upgrade error, the program is lost irreversibly (there is not a bootloader) and the device must be sent back to Lika Electronic for restoring.

11. download progress bars and additional information are shown in the page during operation;

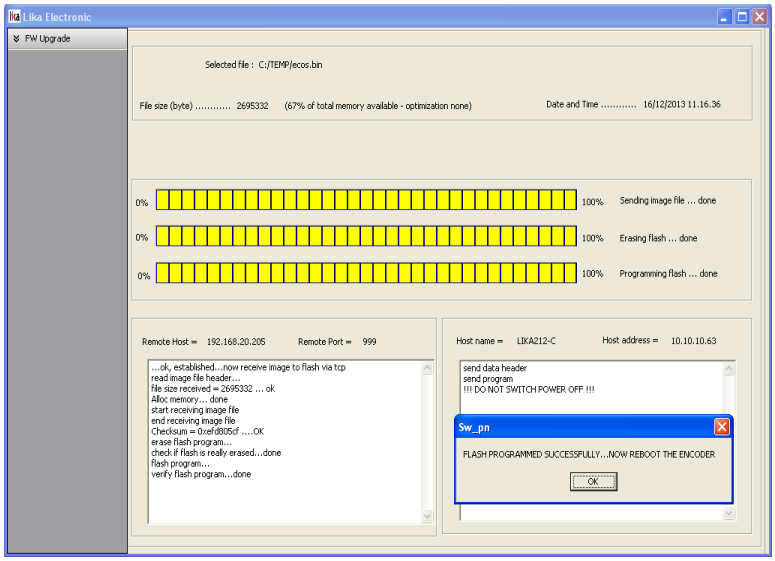


Figure 68 - Firmware upgrade operation process

12. during download operation S1 Status 1 LED starts blinking green at 1 Hz while S2 Status 2 LED is OFF (see on page 34);
13. as soon as the operation is carried out successfully, the FLASH PROGRAMMED SUCCESSFULLY message appears on the screen;
14. turn encoder power off and then on to complete the operation.

NOTE

While downloading the firmware upgrading program, unexpected conditions may arise which could lead to a failure of the installation process. When such a matter occurs, download process cannot be carried out successfully and thus the operation is aborted; S2 Status 2 LED starts blinking red at 1 Hz while S1 Status 1 LED is OFF (see on page 34). In case of flash upgrade error, the program is lost irreversibly (there is not a bootloader) and the device must be sent back to Lika Electronic for restoring.

15.2 Setting the preset value

NOTE

This page is designed to help you setting a Preset value easily. Please note it allows you just to enter and save the value, not to execute the preset. If you need to execute the preset you must set the bit 12 (**Request set/shift of home position**) of the **G1_STW** control word (see on page 75). Refer also to the index **P65000 – Preset value** on page 83; to G1_XIST1 Preset control on page 89; and to the "14.2 Preset diagram" section on page 155. In a customary way you should always use the asynchronous transmission to set the preset value.
Using this page the preset value is stored automatically on the non volatile memory (you do not need to use the parameter **P971 – Transfer to non volatile memory**).

To enter a preset value please proceed as follows:

1. open the Profinet encoder web server by typing the IP address of the encoder you want to connect to (in the example: 192.168.20.195) in the address bar of your web browser and confirm by pressing **ENTER**;

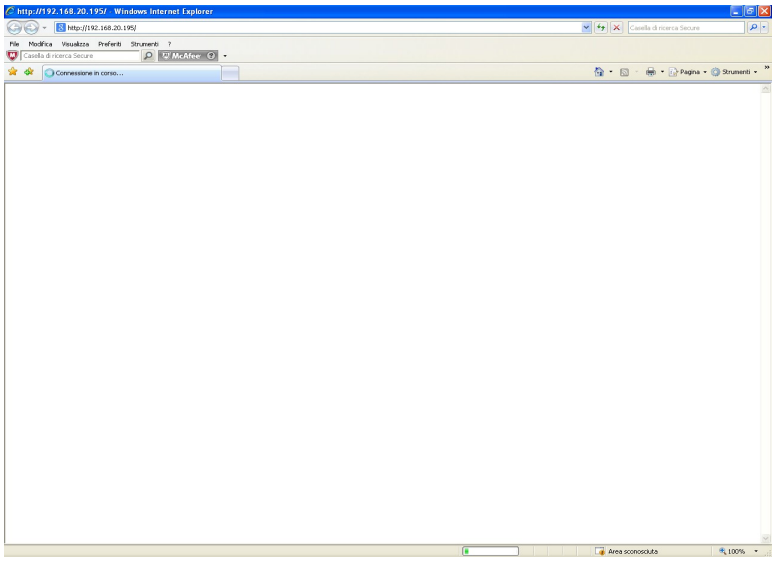


Figure 69 - Opening the web server

2. in the Home page press the **Change the preset value** command to enter the **Change the preset value** page;

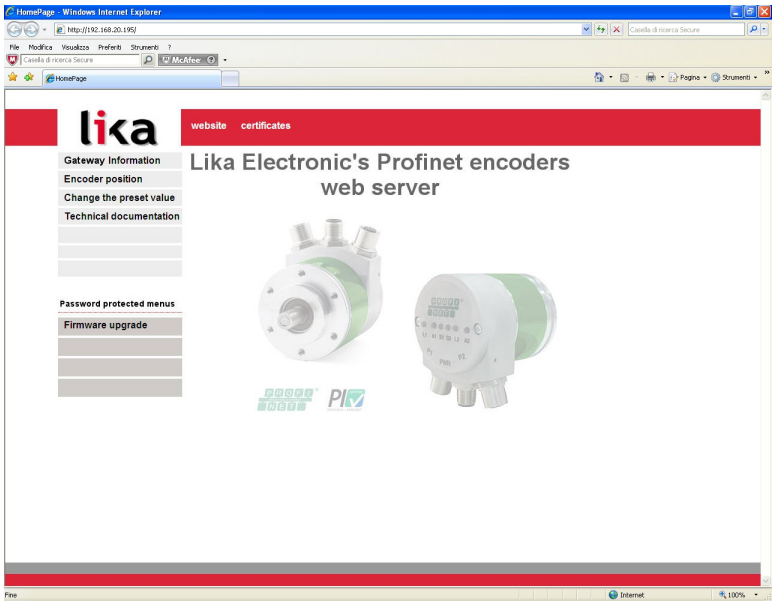


Figure 70 - Web server Home page

3. before entering the page you are requested to confirm the operation; press **OK** to continue;

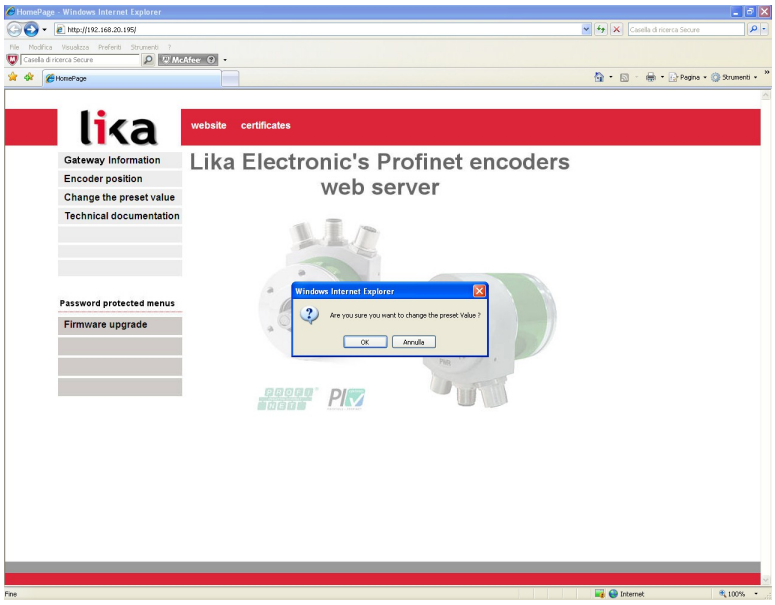


Figure 71 - Confirming the access to the Preset page

4. in the page that appears you have to enter the desired preset value expressed in hexadecimal format; please enter the value without the 0x prefix. Press **ENTER** to confirm. In the example, the value 0x1000 hex (= 4,096 dec) is entered. The preset value is stored automatically on the non volatile memory (you do not need to use the parameter **P971 – Transfer to non volatile memory**);

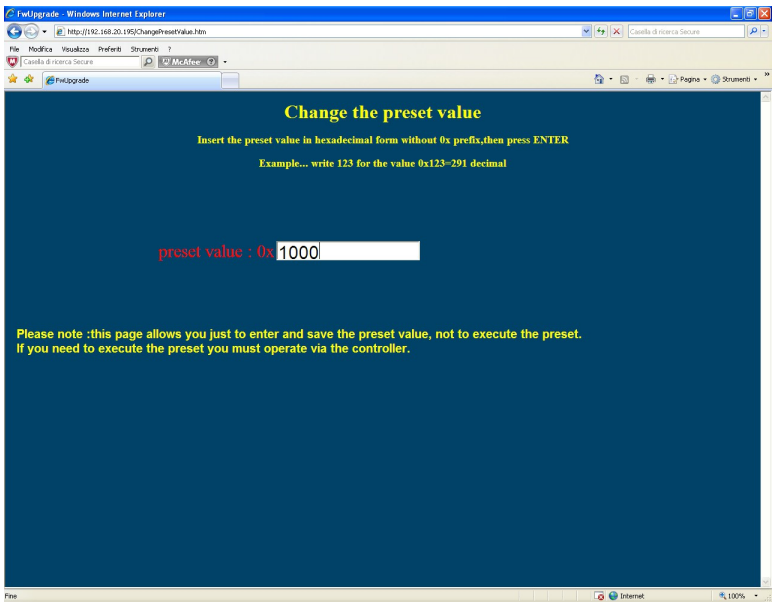


Figure 72 - Changing the Preset value

5. after completing the operation the following page will appear: the preset value has been stored properly on the non volatile memory.

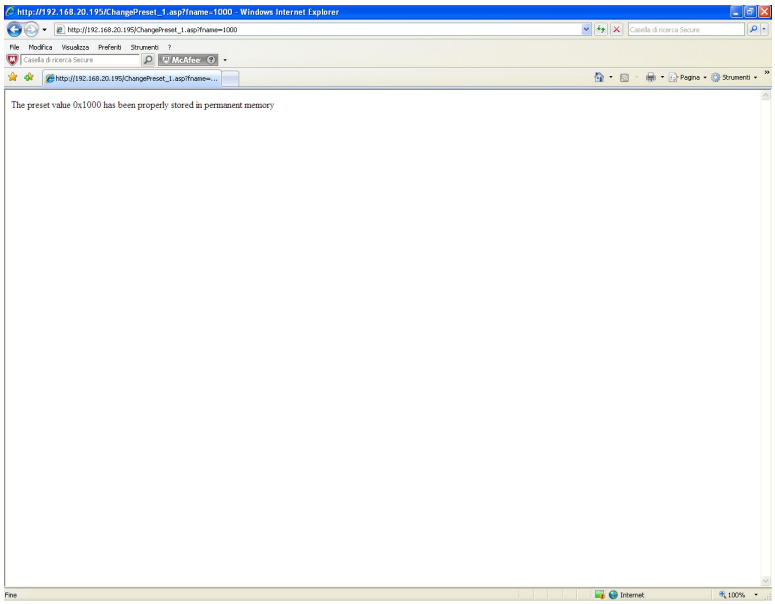


Figure 73 - Preset value stored properly

6. if you try to enter a preset value that is already stored on the memory the following message will be returned.

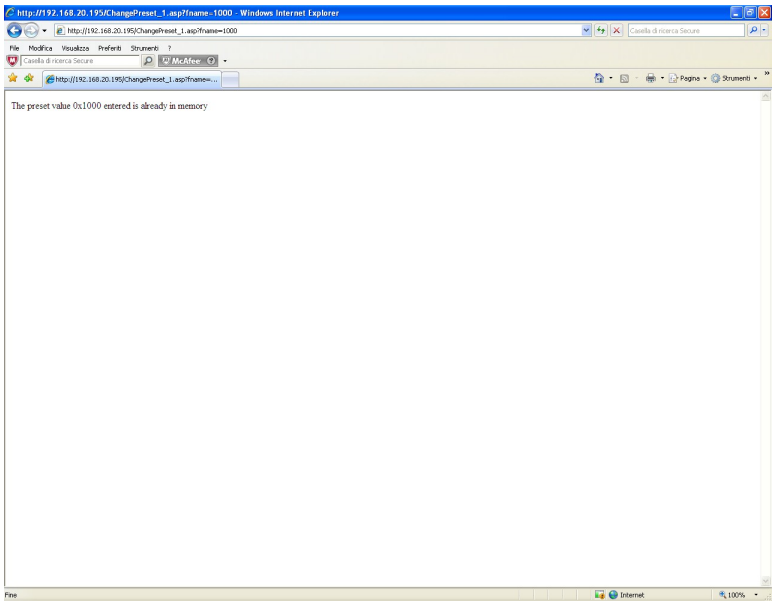


Figure 74 - Preset value already stored

16 Default parameters list

16.1 List of the common parameters

For a comprehensive description of the the listed parameters refer to the "9.4.4 Description of the common parameters" section on page 88.

Parameters list	Default value		
Code sequence	0		
Class 4 functionality	1		
G1_XIST1 Preset control	0		
Scaling function control	0		
Alarm channel control	0		
Compatibility Mode	1		
Maximum tolerated failures of Master Sign-Of-Life	1		
Code Format	0		
Alignment	0		
Bypass	0		
Number of clocks	32		

16.2 List of the parameters specific to linear encoders (DAP 1)

For a comprehensive description of the the listed parameters refer to the "9.4.5 Description of the parameters specific to the linear encoders (DAP1)" section on page 95.

Parameters list	Default value		
Physical pulse resolution [nm]	5,000		
Programmable pulse resolution [nm]	5,000		
Physical Total resolution [bit]	30		
Programmable Total resolution [pulse]	1,073,741,824		

16.3 List of the parameters specific to rotary encoders (DAP 2)

For a comprehensive description of the the listed parameters refer to the "9.4.6 Description of the parameters specific to the rotary encoders (DAP2)" section on page 105.

Parameters list	Default value		
Physical singleturn resolution [bit]	16		
Physical multiturn resolution [bit]	14		
Programmable pulse/rev	65,536		
Programmable total measuring range (pulse)	1,073,741,824		

This page intentionally left blank

This page intentionally left blank

This page intentionally left blank

Document release	Release date	Description	HW	SW	GSDML file version
1.0	09.04.2018	First issue	1	1.0.1	Release 20180319
1.1	27.09.2019	New firmware, new GSDML file, bypass function added and related parameters updated, setting range updated in some parameters, new POWER SUPPLY DIP switch	4	2.0.2	From release 20181129 to ...

Dispose separately

Lika Electronic
Via S. Lorenzo, 25 • 36010 Carrè (VI) • Italy

Tel. +39 0445 806600
Fax +39 0445 806699

info@lika.biz • www.lika.biz