EXAG

Absolute Rotary Encoder

with PROFINET-IO-Interface

Technical manual

ABSOLUTE ROTARY ENCODER WITH PROFINET INTERFACE

USER MANUAL

Content

ABSOLUTE ROTARY ENCODER WITH PROFINET INTERFACE

USER MANUAL

Alterations of specifications reserved

Technical specifications, which are described in this manual, are subject to change due to our permanent strive to improve our products.

Version date:	February 1, 2017
Version number:	2.11
Article number:	10019581
Author:	Reiner Bätjer

1. Introduction

This manual describes the implementation and configuration of an absolute rotary encoder with PROFINET interface. The device fulfills the requirements of a PROFINET IO device with RT (real time) or IRT (isochronous real time) classification.

1.1 Absolute rotary encoder

The basic principle of an absolute rotary encoder is the optical sampling of a transparent code disc in case of an optical encoder or the evaluation of a turning magnet in case of a magnetic encoder which are fixed at the driving shaft.

The absolute rotary encoder has a maximum resolution of 65,536 steps per revolution (16 bits).

The multi-turn versions can detect up to 16,384 revolutions (14 bits). Therefore the largest resulting resolution is 30 bits = 2^{30} = 1,073,741,824 steps. The standard single-turn version has 13 bits, the standard multi-turn version 25 bits.

For further information about the function principle or the setup of a PROFINET network please, refer to http://www.profibus.com/pn.

1.2 PROFINET technology

PROFINET is an Industrial Ethernet standard merging plant automation with other enterprise IT resources.
It provides comparable functionality to PROFIBUS with techniques used by engineering, IT, and management personnel.
Established IT standards are employed as basis of communication: TCP, UDP, IP. XML is used as description language for device profiles (GSDML files).
Two ways of using PROFINET are available: PROFINET IO, similar to PROFIBUS DP as a distributed I/O system and PROFINET CBA as a modular component-based system for larger systems.
PROFINET offers scalable communication for different applications in industrial automation:

- PROFINET NRT (non real time) is suited for non-time-critical process automation with clock rates of roughly 100 msec.
- PROFINET RT (real time) offers a communication channel with optimized performance (10 msec clock rate) for most factory automation tasks
- PROFINET IRT (isochronous real time) employs special communication hardware to enable clock rates of less than 1 msec and a jitter precision of less than 1 µsec. This channel is mainly of use for motion control applications.

PROFINET IO uses a view of distributed I/O similar to PROFIBUS DP. IO controllers (e.g. PLCs) run an automation program, IO devices (e.g. absolute encoders) are remotely assigned field devices, and IO supervisors (e.g. programming devices) are used for commissioning and diagnostics.

The engineering of PROFINET IO is done similar to PROFIBUS. The field buses (i.e. Ethernet topologies) are assigned to control systems during configuration. The IO device is configured in the actual system based on the contents of its GSDML file.
After completion of the engineering the installer loads the data for the expansion into the IO controller (PLC) and the IO controller assumes data exchange with the IO device.

An IO device is addressed within PROFINET (and also possibly by external IT components) through its IP address.

Data can be exchanged from the IO controller to the IO device (and vice versa) cyclically (for process data). Apart from this, parameter data can be exchanged acyclically during engineering of the IO device or by the use of PLC programming blocks.

1.3 Features of the Encoder

- Integrated Boot loader for customer firmware upgrades
- Round axis (Endless shaft)
- Neighbouring detection
- Engineering identification call
- Different filters for velocity
- Used Profinet Encoder Profile V4.0/V4.1

2. Installation

2.1 Electrical Connection

The rotary encoder is connected by a 4 pin M12 connector for the power supply and two 4 pin, D-Coded M12 connector for Ethernet.

The Encoder uses a second D-coded connector and provides integrated switch functionality. On or in the packaging of the connector is the mounting description.

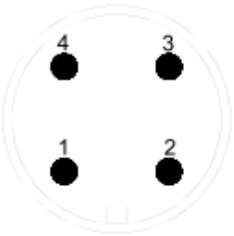
Connector Ethernet
4 pin female, D-coded

Pin Number	Signal
1	Tx +
2	Rx +
3	Tx -
4	Rx -

Connector power supply
4 pin male, A-coded

Pin Number	Signal
1	US (10 - 30 V DC)
2	N.C.
3	GND (0V)
4	N.C.

Sketch on encoder view



2.2 Ethernet cables

2.2.1 RJ45 – M12 crossed

Signal	RJ45 Pin	M12 Pin
Tx+	1	2
Tx-	2	4
Rx+	3	1
Rx-	6	3

2.2.2 RJ45 – M12 straight

Signal	RJ45 Pin	M12 Pin
Tx+	1	1
Tx-	2	3
Rx+	3	2
Rx-	6	4

2.2.3 M12 – M12 crossed

Signal	M12 Pin	M12 Pin
Tx+	1	1
Tx-	2	2
Rx+	3	3
Rx-	4	4

2.3 Diagnostic LEDs

LED	Color	Description for LED = on
Active1	Yellow	Incoming and outgoing traffic at port 1
Link1*	Green	Link to another Ethernet component via port 1
Active2	Yellow	Incoming and outgoing traffic at port 2
Link2*	Green	Link to another Ethernet component via port 2
Stat1	Green	Status 1, details see next table
Stat2	Red	Status 2, details see next table

* Flashes with 2Hz if engineering identification call is activated and link connection is available

2.4 Status LED indication

Status 1 Green	Status 2 Red (Bus failure)	Meaning	Cause
Off	Off	No power	Fuse blown or cable defect
On	On	No connection to controller Criteria: no data exchange	- Bus disconnected - IO-Controller not available / switched off / not in run
On	Blinking 1)	Parameterization fault, no data exchange Criteria: connection available. However, the slave did not switch to the data exchange mode.	- Slave not configured yet or wrong configuration - Wrong station address assigned (but not outside the permitted range) - Actual configuration of the slave differs from the nominal configuration
On	Off	Data exchange. Slave and operation ok.	

1) The blinking frequency is 0.5 Hz. Minimal indication time is 3 sec.

2.5 Instructions for mechanical installation and electrical connection of the rotary encoder

The following points should be observed:

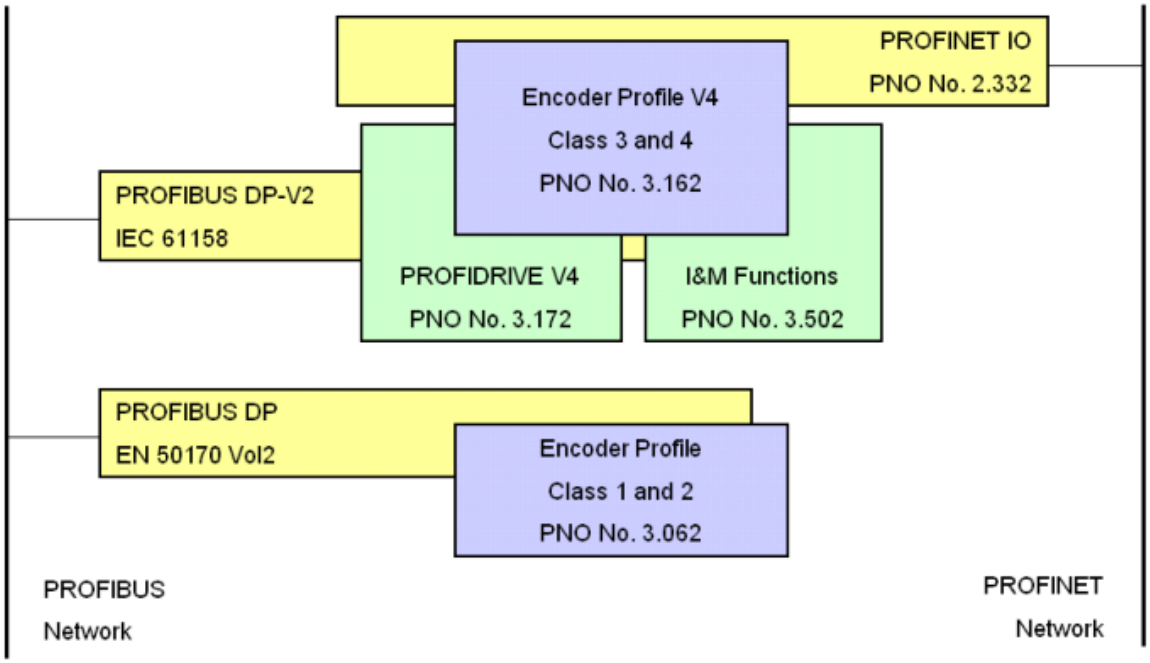
- Do not drop the angular encoder or subject it to excessive vibration. The encoder is a precision device.
- Do not open the angular encoder housing. If the device is opened and closed again, it can be damaged and dirt may enter the unit.
- The angular encoder shaft must be connected to the shaft to be measured through a suitable coupling (full shaft version). This coupling is used to dampen vibrations and imbalance on the encoder shaft and to avoid inadmissible high forces. Suitable couplings are available from Posital.
- Although Posital absolute encoders are rugged, when used in tough ambient conditions, they should be protected against damage using suitable protective measures. The encoder should not be used as handles or steps.
- Only qualified personnel may commission and operate these devices. These are personnel who are authorized to commission, ground and tag devices, systems and circuits according to the current state of safety technology.
- It is not permissible to make any electrical changes to the encoder.
- Route the connecting cable to the angular encoder at a considerable distance or completely separated from power cables with their associated noise. Completely shielded cables must be used for reliable data transfer and good grounding must be provided. Cabling, establishing and interrupting electrical connections may only be carried-out when the equipment is in a no-voltage condition. Short-circuits, voltage spikes etc. can result in erroneous functions and uncontrolled statuses which can even include severe personnel injury and material damage.
- The encoder should have got a large-area connection to PE. If the flange don't have a good electrical connection to the machine – i.e. if there was used a plastic mounting device – then use i.e. a 30cm long and 2cm wide copper tape to get the PE connection.

Before powering-up the system, check all of the electrical connections. Connections, which are not correct, can cause the system to function incorrectly. Fault connections can result in severe personnel injury and material damage.

3. Device configuration

3.1 Standardization

This actual generation of PROFINET devices is based on the Encoder Profile V4.0/V4.1 (PNO No. 3.162). With this standardization it is possible to substitute all products that fulfill the specification.
See the next figure with the coherences.



3.2 Encoder Classes

Application Class	Description
3	Isochronous mode **is not** supported (RT)
4	Isochronous mode **is** supported (IRT)

3.3 Encoder functions

Function	Implementation	
	Class 3	Class 4
Code sequence	-/✓*	✓
Class 4 functionality	✓	✓
G1_XIST1 Preset control	-/✓*	✓
Scaling function control	-/✓*	✓
Alarm channel control	✓	✓
Preset value	-/✓*	✓
Preset value 64bit	-	-
Measuring units per revolution / Measuring step	-/✓*	✓
Total measuring range	-/✓*	✓
Measuring units per revolution 64bit	-/✓*	✓
Total measuring range 64bit	-/✓*	✓
Maximum Master Sign-Of-Life failures	-/✓*	✓
Velocity measuring unit	-/✓*	✓
Encoder Profile version	✓	✓
Operating time	-	-
Offset value	-/✓*	✓
Offset value 64 bit	-/✓*	✓
Round axis (Endless shaft)	✓	✓
Velocity filter	✓	✓

* If Class 4 functionality is activated

3.4 Signal list for Cyclic Data Transmission

Signal No.	Significance	Abbreviation	Length (bit)	Sign
3	Master's sign-of-life	STW2_ENC	16	-
4	Slave's sign of life	ZSW2_ENC	16	-
6	Velocity value A	NIST_A	16	✓
8	Velocity value B	NIST_B	32	✓
9	Control word	G1_STW	16	-
10	Status word	G1_ZSW	16	-
11	Position value 1	G1_XIST1	32	-
12	Position value 2	G1_XIST2	32	-
39	Position value 3	G1_XIST3	64	-

3.4.1 Format of actual position values

G1_XIST1 and G1_XIST2 are the actual position values in binary. For absolute encoders one format example is given below. **NOTE:** the alignment in the data-frame (left or right-aligned) is considered for each individual resolution.

Example: 25 bit Multi-turn absolute encoder (8192 steps per revolution, 4096 distinguishable revolutions).

- All values are presented in binary format G1_XIST2 displays the error telegram instead of the right aligned position value if error occurs.
- The shifting factors in P979 "sensor format" display the actual format. P979, Subindex 4 (Shift factor for G1_XIST2) = 0
- The settings in the Encoder parameter data affect the position value in both G1_XIST1 and G1_XIST2.

Case Encoder Profile 4.0*

- The default setting is G1_XIST1 left aligned.
- P979, Subindex 3 (Shift factor for G1_XIST1) = 32 – Total resolution (next binary value)
- G1_XIST1 send values independent Bit 10 in stw2 and Bit 13 in g1_stw1

Case Encoder Profile 4.1*

- The default setting is G1_XIST1 right aligned.
- If a 32bit counter that starts with the absolute position value. After increasing maximum counter value start again with 0 or after 0 decreasing to the maximum counter value
- P979, Subindex 3 (Shift factor for G1_XIST1) = 0
- G1_XIST1 send values independent Bit 10 in stw2 and Bit 13 in g1_stw1

* Profile 4.0 realized with GSDML-V2.2-SACNCON-SAG-20100808, Profile 4.1 with newer files

M = Distinguishable Revolutions (Multi-turn value)
S = Pulses (Single-turn steps per revolution)

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 value in G1_XIST1 for Encoder Profile 4.0

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	M	M	M	M	M	M	M	S	S	S	S	S	S	S	S	S	S	S	S	S

"Absolute value" in G1_XIST1 for Encoder Profile 4.1

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 value in G1_XIST2

G1_XIST3

For 64bit position values is the G1_XIST3 available. The binary value will transmit right aligned and without shifting factor.

IO Data	1	2	3	4
Format	64 bit position value			

3.4.2 Encoder control word (STW2_ENC)

4-Bit-counter, left justified. The master application starts the sign of life with any value between 1 and 15. The master increases the counter in every cycle of the master application.

Valid values for the master's sign of life are 1 to 15, "0" indicates an error and is left out in normal operation.

		Implementation	
Bit	Function	Class 3	Class 4
0…9	Reserved, currently not used		
10	Control by PLC	✓	✓
11	Reserved, currently not used		
12…15	Controller Sign-Of-Life	-	✓

Bit	Value	Significance	Comments
10	1	Control by PLC	Control via interface, EO IO Data is valid
	0	No control by PLC	EO IO Data is not valid. Except Sign-Of-Life
12…15		Controller Sign-Of-Life	Send continuous counting value from 0 to 15

3.4.3 Encoder status word (ZSW2_ENC)

4-Bit-counter, left justified. The slave application starts the sign of life with any value between 1 and 15 after successful synchronization to the clock pulse. The counter is increased by the slave application in every DP-cycle. Valid values for the slave's sign of life are 1 to 15, "0" indicates an error and is left out in normal operation.

		Implementation	
Bit	Function	Class 3	Class 4
0…8	Reserved, currently not used		
9	Control requested	Mandatory	Mandatory
10…11	Reserved, currently not used		
12…15	Encoder Sign-Of-Life	-	Mandatory

Bit	Value	Significance	Comments
9	1	Control requested	The automation system is requested to assume control
	0	No control by PLC	EO IO Data is not valid. Except Sign-Of-Life
12…15		Encoder Sign-Of-Life	Send back continuous Controller Sign-Of-Life (counting value from 0 to 15)

3.4.4 Encoder control word (G1_STW)

Bit	Value	Function	Comments
0.. 10			Reserved, currently not used
11	0/1	„Home position mode"	Specifies if the position value shall be set to a previously programmed absolute value or shifted by this value. 0: set home position / preset (absolute) 1: shift home position / preset (relative = offset)
12	1	Set preset / request shift	Preset (resp. shift) is set when changing this Bit to "1" (rising edge). Default preset value (shift): 0 **Warning:** After setting the preset the offset will be save in the non volatile memory. In this 5-10ms the encoder will not send position values.
13	1	Request absolute value cyclically	Request of additional cyclic transmission of the absolute actual position in G1_XIST2. If no other data needs to be transferred due to commands or errors the absolute position value will be transmitted automatically.
14	1	Activate parking sensor	If the "activate parking sensor" bit is set, the encoder transmits no error messages.
15	1	Acknowledging a sensor error	Request to acknowledge / reset a sensor error

3.4.5 Encoder status word (G1_ZSW)

Bit	Value	Meaning	Comment
0 .. 10			Reserved, currently not used
11		Acknowledgement sensor error in process	Is set if the reset of a sensor error (after acknowledging) takes longer than one bus cycle.
12	1	Set preset / shift reference point executed	Acknowledgement for "set preset / request shift"
13	1	Transmit absolute value cyclically	Acknowledgement for "request absolute value cyclically"
14	1	Parking sensor activated	Acknowledgement for "activate parking sensor". The encoder transmits no error messages.
15	1	Sensor error	Indicates a sensor error. A device specific error code is transmitted in G1_XIST2.

3.5 Standard + manufacture telegrams

Standard Telegram 81

IO Data (DWord)	1	2
Setpoint	STW2_ENC*	G1_STW1*

* Details about the variables are available in chapter 3.4

IO Data (DWord)	1	2	3	4	5	6
Actual value	ZSW2_ENC*	G1_ZSW1*	G1_XIST1*		G1_XIST2*	

Standard Telegram 82

IO Data (DWord)	1	2
Setpoint	STW2_ENC*	G1_STW1*

IO Data (DWord)	1	2	3	4	5	6	7
Actual value	ZSW2_ENC*	G1_ZSW1*	G1_XIST1*		G1_XIST2*		NIST_A*

Standard Telegram 83

IO Data (DWord)	1	2
Setpoint	STW2_ENC*	G1_STW1*

IO Data (DWord)	1	2	3	4	5	6	7	8
Actual value	ZSW2_ENC*	G1_ZSW1*	G1_XIST1*		G1_XIST2*		NIST_B*	

Standard Telegram 84

IO Data (DWord)	1	2
Setpoint	STW2_ENC*	G1_STW1*

IO Data (DWord)	1	2	3	4	5	6	7	8	9	10
Actual value	ZSW2_ENC*	G1_ZSW1*	G1_XIST3*				G1_XIST2*		NIST_B*	

Manufacture Telegram 860

With this telegram it is not necessary to set special bits to get cyclic data transmission. It is ajar according the Profibus functionality and support an easy way to set a customer preset value during the running PLC. The velocity value uses the format that is defined in the Velocity measuring unit.

- No control word
- No Status word
- No Life Sign monitoring.
- Output: 32 Bit-Unsigned Preset value (Bit 31 Preset-Control , less than Total Resolution)
- Input: 32 Bit-Unsigned Position Value + 32 Bit-Integer Velocity Value

Input Data (Input data from Encoder to Controller): 8 Bytes

Position value - 32 Bit Unsigned Integer				Velocity value - 32 Bit Signed Integer			
Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
MSB			LSB	MSB			LSB

Output Data (Output data from Controller to Encoder): 4 Bytes

Preset - 32 Bit Unsigned Integer	
Bit 31	Bit 30 .....Bit 0
Preset Control	Preset value < Total Resolution

3.6 Configuration principle

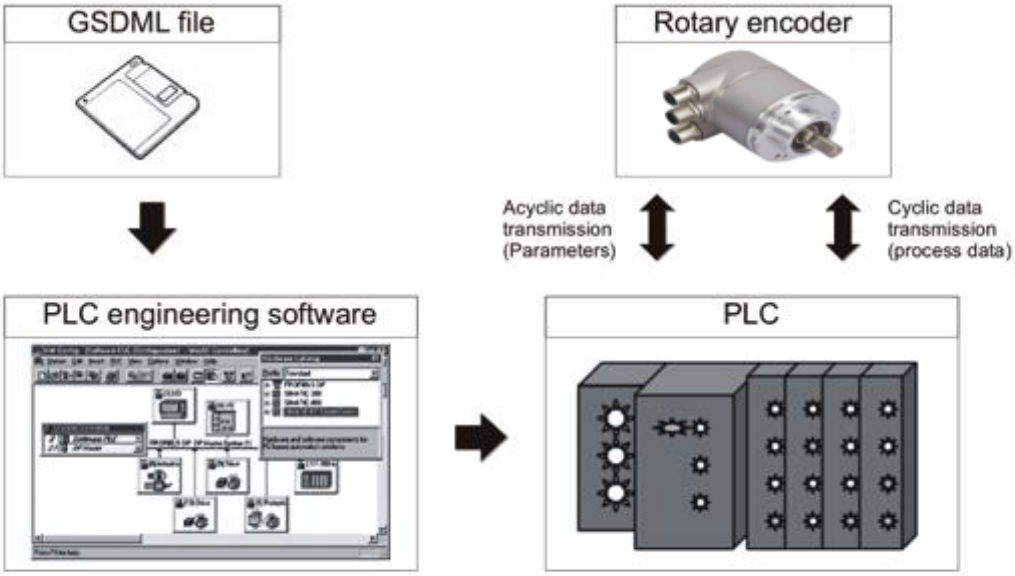
The rotary encoder with PROFINET interface can be programmed according to the needs of the user. The GSDML file pertaining to the rotary encoder has to be installed in the used PLC engineering software tool.

3.7 Rotary encoder functionality overview

Function	Communication channel
Position value	Cyclic input (IO device -> IO controller)
Preset	Cyclic output (IO controller -> IO device)
Coding sequence	Acyclic input/output
Scaling function	Acyclic input/output

3.8 Rotary encoder functions – data format

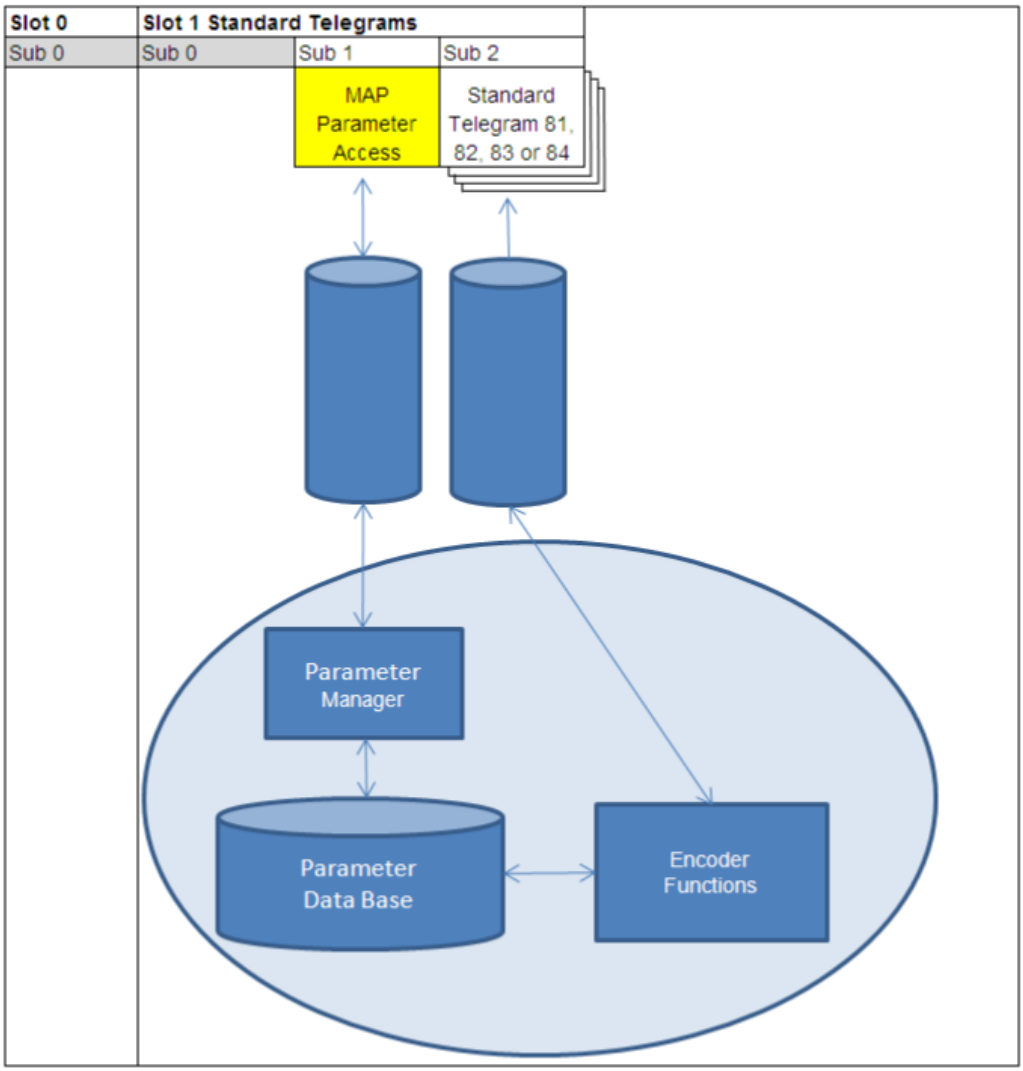
PROFINET IO devices are set up in modules. Each module can be plugged in physical and/or logical slots. These are subdivided into sub slots individually to accommodate further data hierarchy. One sub slot can contain several cyclic input/output channels as well as acyclic record channels (used for parameters).
There are two versions of PLC available. Some of them support only one sub slot. Other ones i.e. S7 400 support several sub slots. To work with both PLCs there are in the GSDML-file two directories: Standard and Encoder Profile 4.
SCANCON rotary encoders offer for the standard profile one slot (address #0) with one sub slot (address #0) for all device data for old PLC's that doesn't support several sub slots.
Device parameters are grouped together as records in the PROFINET interface. The following table gives an overview over addresses of SCANCON rotary encoder's data channels.



3.9 Parameter for Acyclic Data Transmission

The user parameter data is sent to the encoder in the start-up phase as a Record Data Object using the data record 0xBF00. For the mapping of the different encoder functions into the user data section of the Record Data Object. In addition to the parameter data configuration the encoder supports a number of PROFIdrive parameters and encoder specific parameters accessible via the Acyclic Data Exchange service.

Beginning with GSDML version GSDML-V2.2-SCANCON-SAG-20100808 it is possible to change the telegram type without changing the MAP parameters.



3.9.1 Base Mode Parameter

Function	Slot	Sub slot	Index	Offset	Length	IO
Code sequence	1	1	0xBF00	0.0	1 Bit	-
Class 4 functionality	1	1	0xBF00	0.1	1 Bit	-
G1_XIST1 Preset control	1	1	0xBF00	0.2	1 Bit	-
Scaling function control	1	1	0xBF00	0.3	1 Bit	-
Alarm channel control	1	1	0xBF00	0.4	1 Bit	-
Compatibility mode	1	1	0xBF00	0.5	1 Bit	-
Measuring units per revolution	1	1	0xBF00	1	8 Byte	-
Total measuring range	1	1	0xBF00	9	8 Byte	-
Maximum Master Sign-Of-Life failures	1	1	0xBF00	17	1 Byte	-
Velocity measuring unit	1	1	0xBF00	18	1 Byte	-

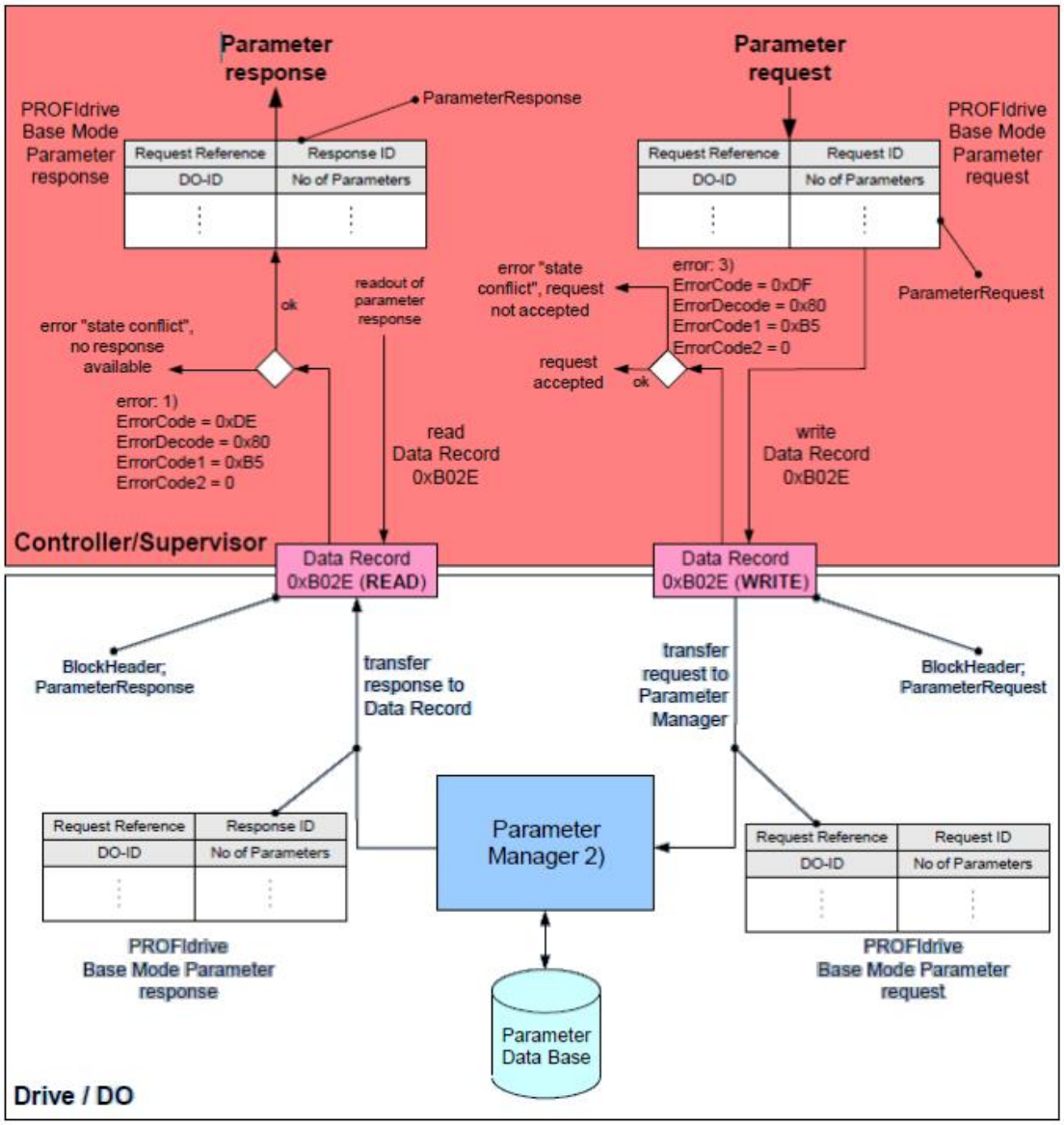
3.9.2 Device Parameter

Function	Slot	Sub slot	Index	Offset	Length	IO
Preset value	1	1	0xB02E	Via Parameter Number 65000		-

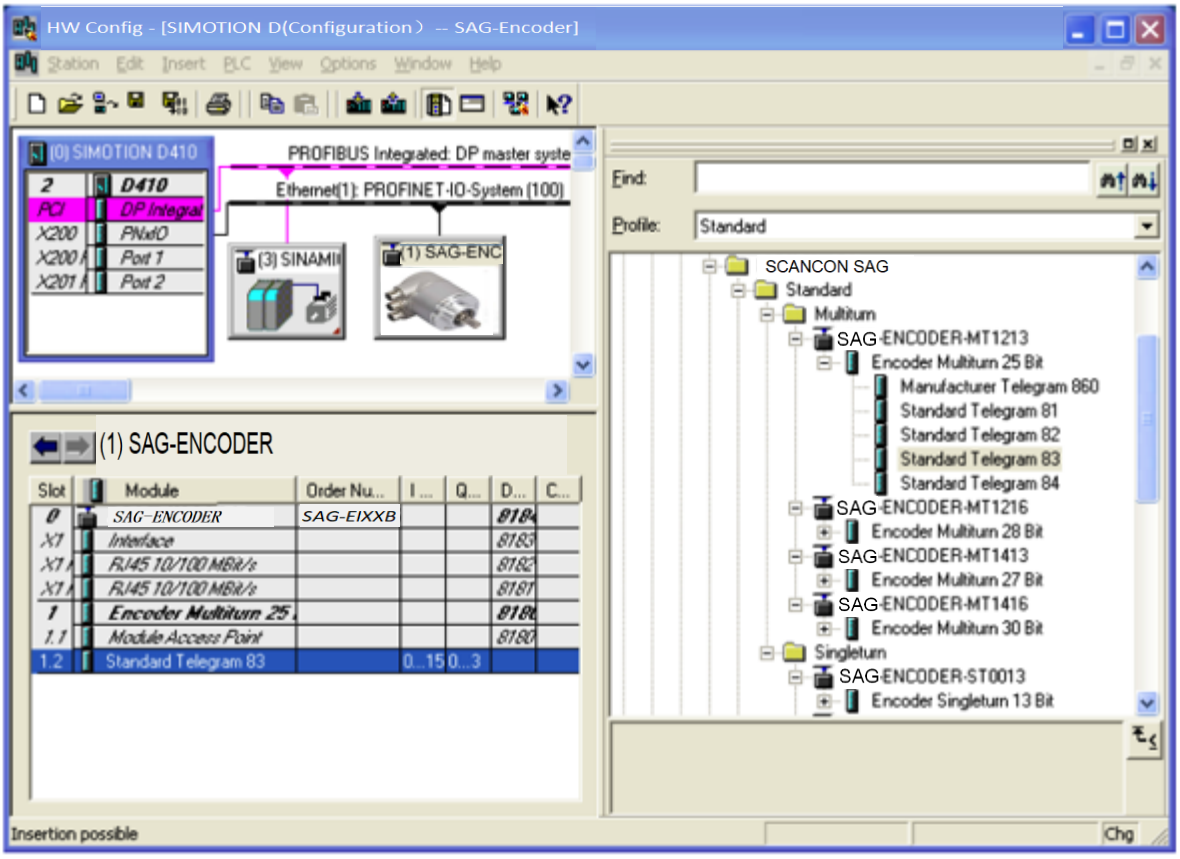
3.9.3 Vendor Parameter

Function	Slot	Sub slot	Index	Offset	Length	IO
Velocity filter	1	1	0x1000	0	1 Byte	-

Parameter model



Sample of configuration according Encoder Profile V4.1



3.11 Rotary encoder function description

Details of this functionality are available on the next pages.

	Implementation		Description
Function	Class 3	Class 4	Chapter
Code sequence	-/✓*	✓	3.11.1
Class 4 functionality	✓	✓	3.11.2
G1_XIST1 Preset control	-/✓*	✓	3.11.3
Scaling function control	-/✓*	✓	3.11.4
Alarm channel control	✓	✓	3.11.5
Compatibility mode	✓	✓	3.11.6
Preset value	-/✓*	✓	3.11.7
Preset value 64bit	-	-	-
Measuring units per revolution / Measuring step	-/✓*	✓	3.11.9
Total measuring range	-/✓*	✓	3.11.9
Measuring units per revolution 64bit	-/✓*	✓	-
Total measuring range 64bit	-/✓*	✓	-
Maximum Master Sign-Of-Life failures	-/✓*	✓	3.11.10
Velocity measuring unit	-/✓*	✓	3.11.11
Encoder Profile version	✓	✓	3.11.14
Operating time	-	-	-
Offset value	-/✓*	✓	3.11.8
Offset value 64 bit	-/✓*	✓	-
Round axis (Endless shaft)	✓	✓	3.11.13
Velocity filter	✓	✓	3.11.12

* If Class 4 functionality is activated

3.11.1 Code sequence

The parameter "code sequence" defines the counting direction of the position value. The code increases when the shaft is rotating clockwise (CW) or counter-clockwise (CCW) (view onto the shaft).

Code sequence	Direction of rotation when viewing the shaft	Code sequence
0 (default)	Clockwise (CW)	Increasing
1	Counter-clockwise (CCW)	Decreasing

3.11.2 Class 4 functionality

The parameter "Class 4 functionality" defines that the scaling, preset and code sequence affects the position value in G1_XIST1, 2 and 3.

Class 4 control	Class 4 function
0 (default)	Deactivated
1	Activated

3.11.3 Preset control for G1_XIST1

The parameter "preset control" defines the preset functionality. If parameter Class 4 is activated and Preset control is disabled then the Preset will not be affected for G1_XIST1.

Preset control	Preset function
1	Preset does not affect G1_XIST1
0 (default)	G1_XIST1 is affected by a Preset command

3.11.4 Scaling function control

The parameter "scaling function control" enable / disenable the scaling function. If not, the physical position value is returned by the rotary encoder. This is only available if class 4 control is activated.

Scaling function control	Scaling function
0	Deactivated
1 (default)	Activated

3.11.5 Alarm channel control

The parameter "Alarm channel control" defines the length of diagnostic telegram. If the Alarm channel is deactivated then will only transmit the first 6 bytes of the diagnostic telegram.

Alarm channel control	Alarm channel function
0 (default)	Deactivated
1	Activated

3.11.6 Compatibility mode

This parameter defines if the encoder should run in a mode compatible to Version 3.1 of the Encoder Profile. See next tables for an overview of the functions affected when the compatibility mode is enabled.

Compatibility mode	Compatibility function	Meaning
0	Enable	Compatibility with Encoder Profile V3.1
1 (default)	Disable	No backward compatibility

Function	Compatibility mode Enable (=0)	Compatibility mode Enable (=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 Master Sign-Of-Life failures"	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 - Profile version	31 (V3.1)	41 (V4.1)

3.11.7 Preset value

3.11.7.1 Telegram 81-84

With the Preset value it is possible to adapt the encoder zero point to the zero point of the application. When using this function the current encoder position value is set to the desired preset value. The integrated microcontroller calculates the internal zero point shift. It is stored in a permanent memory (~ 10 ms).

- Set Preset only in standstill!
- There is no preset activated when the Preset value is written to the encoder. The preset function is controlled by the bits in sensor control and status words (G1_STW and G1_ZSW). The Preset value is used when a preset is requested by bit 12 in the Sensor control word (G1_STW).
- Class 4 functionality must be enabled!
- If the Preset value is greater than the total resolution then error no. 0x02 comes back to the base mode parameter response (Low or High limit exceeded).

NOTE:

Parameter	Meaning	Data type
Preset value	Preset value will defined with asynchronous data exchange. Default value = 0	Integer 32

Sample for a parameter order to set Preset with Record Read-Write for SIMATIC CPU300.

RecordWriteData[] = {

```
0x00,0x02,0x00,0x01,                // Header
0x10,0x00,0xFD,0xE8,0x00,0x00,      // Parameter Address (Preset)
0x43,0x01,0x00,0x00,0x00,0x64       // Parameter Value (Preset=100=0x64h)
};
```

```
Meaning:
0x00,0x02,0x00,0x01,                // Header
  |    |     |      |------------   No. of Parameters = 1
  |    |     |-------------------   Axis-No./DO-ID = 0
  |    |-------------------------   Request ID      = 2 Change value
  |--------------------------------  Request Reference

0x10,0x00,0xFD,0xE8,0x00,0x00,      // Parameter Address (Preset)
  |     |     |     |     |    |-----  Subindex  LOW Byte
  |     |     |     |     |----------  Subindex HIGH Byte
  |     |     |     |---------------   Parameter Number (PNU) LOW Byte
  |     |     |-------------------     Parameter Number (PNU) HIGH Byte
  |     |-------------------------     No. of Elements
  |--------------------------------    Attribute

0x43,0x01,0x00,0x00,0x00,0x64       // Parameter Value  (Preset Value = 100 = 0x64 Hex)
  |     |     |     |     |    |-----  Preset Value  LSB
  |     |     |     |     |----------  Preset Value
  |     |     |     |---------------   Preset Value
  |     |     |-------------------     Preset Value  MSB
  |     |-------------------------     No. of Values =1
  |--------------------------------    Format  : 0x43= DWORD , oder  4= Ingeger 32Bit
```

SIMATIC S7: **-SFB53**
-FC x:

```
CALL  "WRREC" , DB53
 REQ    :=M41.7          // activate sfb request
 ID     :=DW#16#0        // logical slot address -> adapt
 INDEX  :=W#16#B02E      // record index number  sizeof(RecordWriteData)
 LEN    := 16            // data length in byte
 DONE   :=M41.1          // request finished
 BUSY   :=M41.2          // busy bit
 ERROR  :=M41.3          // error bit
 STATUS:=MD46            // error number, if error bit = 1
 RECORD:= RecordWriteData[] // record buffer address -> adapt
```

3.11.7.2 Telegram 860

With this manufacture telegram it is easy to set a user defined preset value during the running application according to the Profibus functionality. In this case set bit 31 of the Output Data to "1" and then back to "0". For a different preset value than 0 set the other bits.

Output Data (Output data fromController to Encoder): 4 Bytes	
Preset - 32 Bit Unsigned Integer	
Bit 31	Bit 30 .....Bit 0
Preset Control	Preset value < Total Resolution

If the Preset value is greater than the total resolution then will set the Preset value to the maximum resolution - 1.

3.11.8 Offset value

The offset value is calculated in the preset function and shifts the position value with the calculated value.

3.11.9 Scaling parameters

The Scaling parameters will be used to change the resolution. This parameter will only affect to the output values if the Scaling function is activated.

Parameter	Meaning	Data type
Measuring units per revolution / Measuring step	Single turn resolution in steps	Unsigned 32
Total measuring range in measuring units	Total measuring range measuring steps	Unsigned 32

3.11.10 Max. Master Sign-Of-Life failures

With this parameter the number of allowed failures of the master's sign of life is defined.

Parameter	Meaning	Value
Maximum Master Sign-Of-Life failures	Number of permissible failures of the master's life sign	1 … 255

3.11.11 Velocity measuring units

This parameter defines the coding of velocity measuring units used to configure the values NIST_A and NIST_B. Only Telegrams 82-84 uses the velocity outputs.

With each cycle will calculate the velocity from the position value. To get a high velocity precision it is necessary to use a short cycle time.

Velocity measuring unit	Value
Steps/s	0
Steps/100ms	1
Steps/10ms	2
RPM	3

3.11.12 Velocity filter

The velocity value can used with three different exponential moving average filter types.

Default: Fine

Parameter	Meaning	Data type
Velocity filter	Select for the parameter Fine, Normal, Coarse	Integer 32

Ratio between old and actual velocity value:
Fine: 7:3
Normal: 96:4
Coarse: 996:4

3.11.13 Endless Shaft (RoundAxis)

Normally the period, i.e. "Total resolution" / "measuring units" per revolution must be an integer and it must fit an integer number of times (integer multiple) into 4096 for an encoder with 12 Bit for the revolutions. This means that i.e. 100 or 325 revolutions could make trouble. So the following equation must apply:

(4096 x measuring units per revolution) / Total resolution = integer

But this Profinet encoder solves this problem automatically. The encoder checks if the parameters need the endless shaft and activates the functionality by self.

Note: The internal software routine only works if the encoder is in operation. If it is necessary to turn the encoder shaft more than 1024 revolutions without power supply this can lead to problems (the internal routine will not work without power supply). With this function there will be save additional values in the internal eeprom.

3.11.14 Encoder Profile version

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.

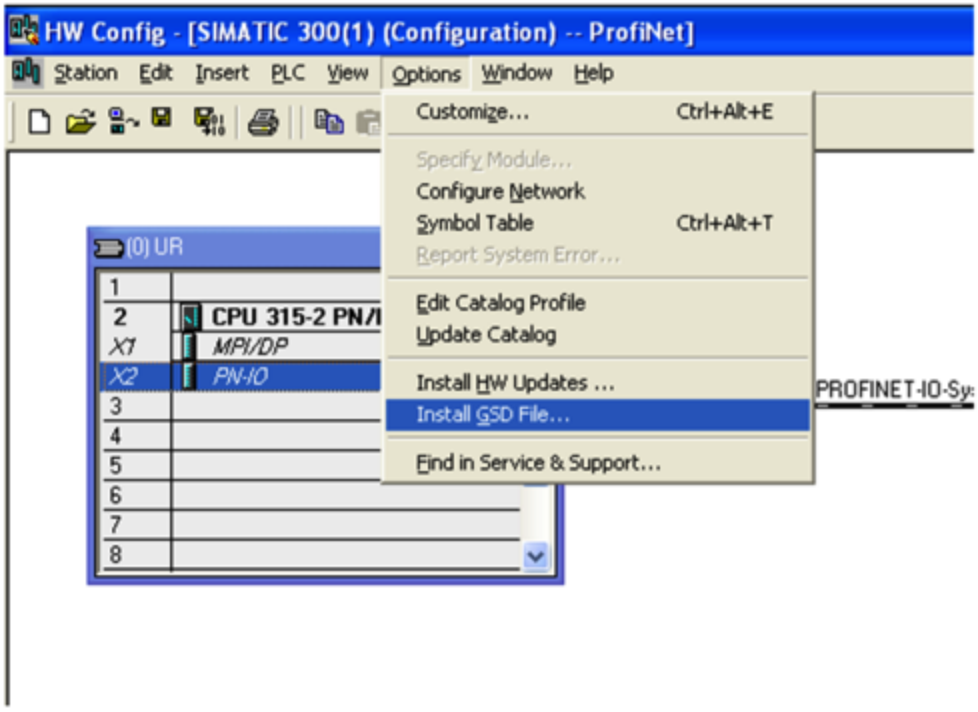
Bits	Meaning
0-7	Profile Version, least significant number (value range: 0…99), decimal coding
8-15	Profile Version, most significant number (value range: 0…99), decimal coding
16-31	Reserved

4. Configuring with STEP7

In the following chapter the configuration of the SCANCON encoder with the configuration tool Hardwaremanager STEP 7 is shown exemplarily. In this example STEP 7 Version 5.4 SP4 and the CPU 315-2PN/DP or Simotion Scout with single axis controller D410 (PROFINET controller integrated) are used. If there are questions about other software tools please contact the manufacturer.

4.1 Installing the GSDML file

If SCANCON encoders are used for the first time it is necessary to install the GSDML file to import encoder parameterization into the hardware catalogue of the tool:
Choose "Install GSD File..." in the "HW Config"-window of the project (menu item "Options") and select the GSDML-file.
The GSDML file is supplied by SCANCON (free of charge from scancon.dk). In order to represent the encoder with a bitmap in STEP7 the bitmap file will be installed automatically with the GSDML file – both files must be in the same directory. The main number of the "Software Release" in the GSDML file and the Firmware must be the same, i.e. 4.xx.



After the successful installation of the GSDML file the SCANCON encoder can be found in the hardware catalog under „PROFINET-IO" – „Additional Field Devices" – „Encoders" – „SCANCON SAG".

(Possibly, you need to update the hardware catalog by choosing "Options" -> "Update catalog").

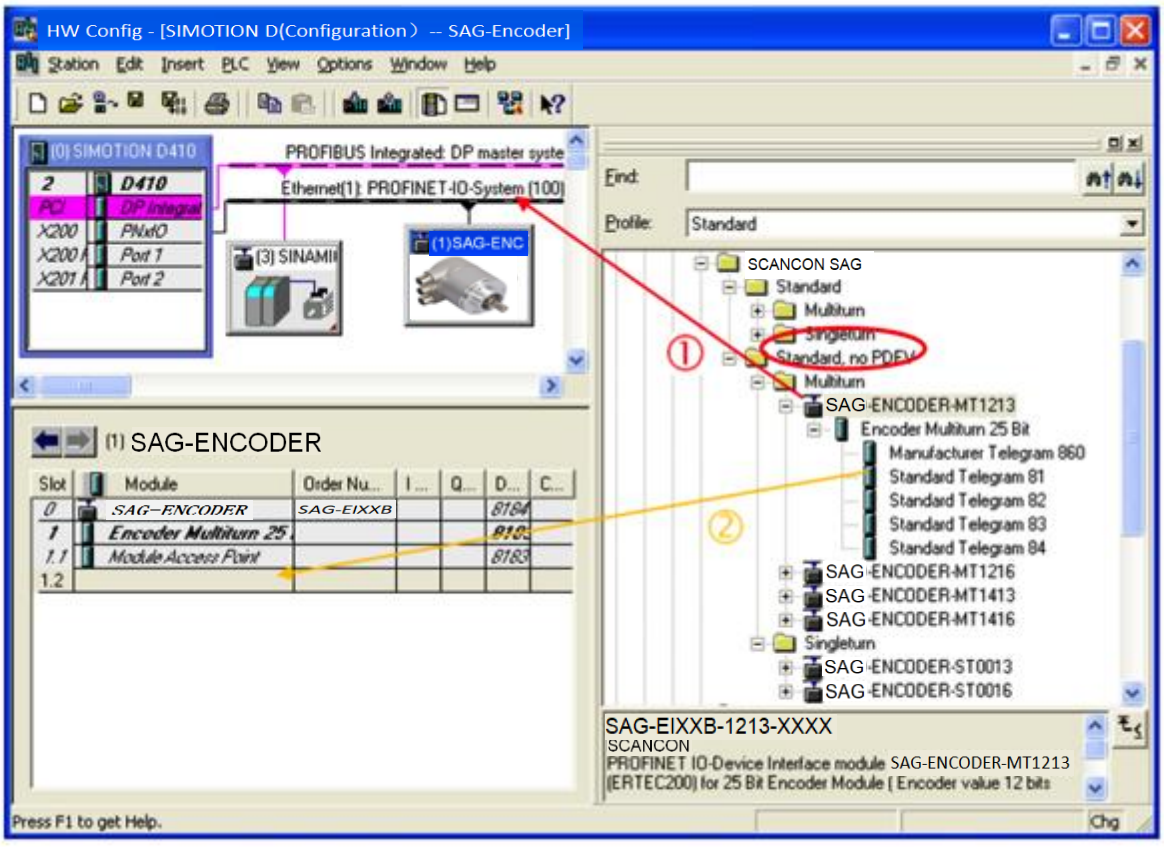
4.2 Engineering a SCANCON encoder into a STEP7 project

To engineer the rotary encoder into a project, drag the device "SAG ENCODER..." on to an existing PROFINET ethernet network (or choose the network and double-click the 'SAG SCANCON' icon).

See the red arrow. Then move the telegram to the free slot (orange arrow).

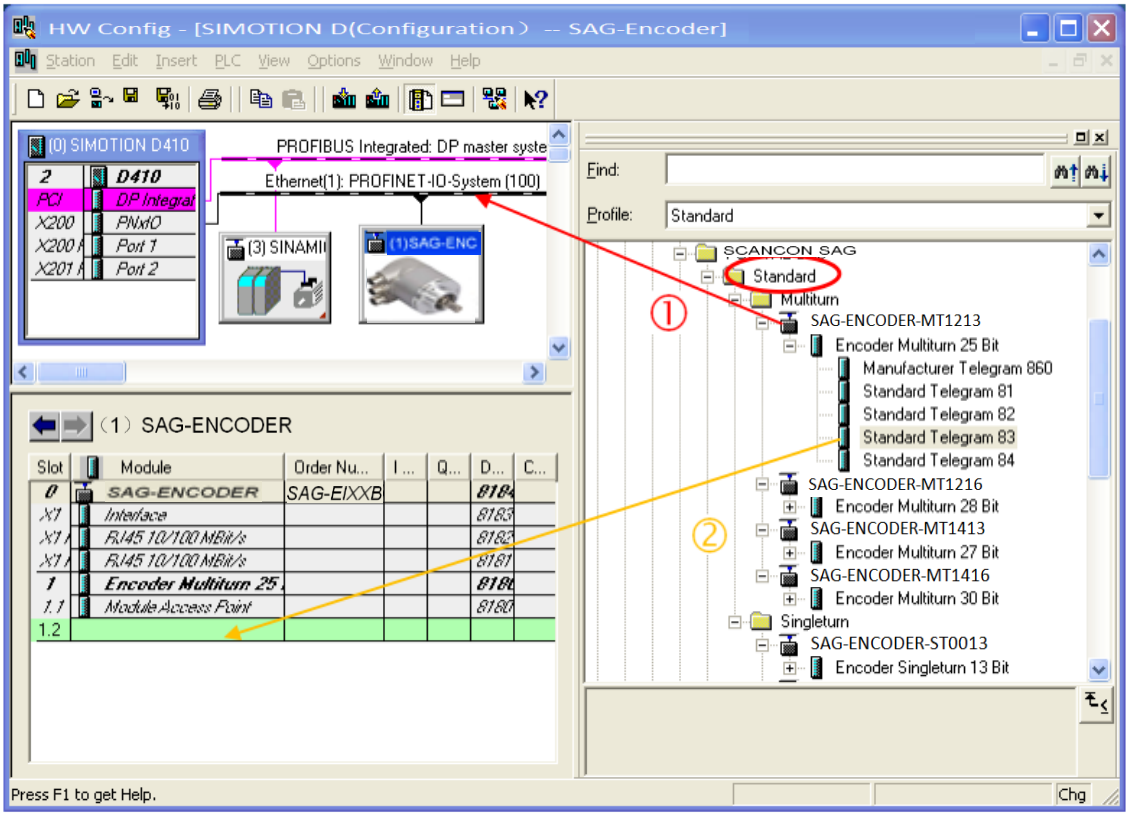
4.2.1 Standard Encoder no PDEV

Asynchronous + RT Communication for Controller which does not support IRT functionality.

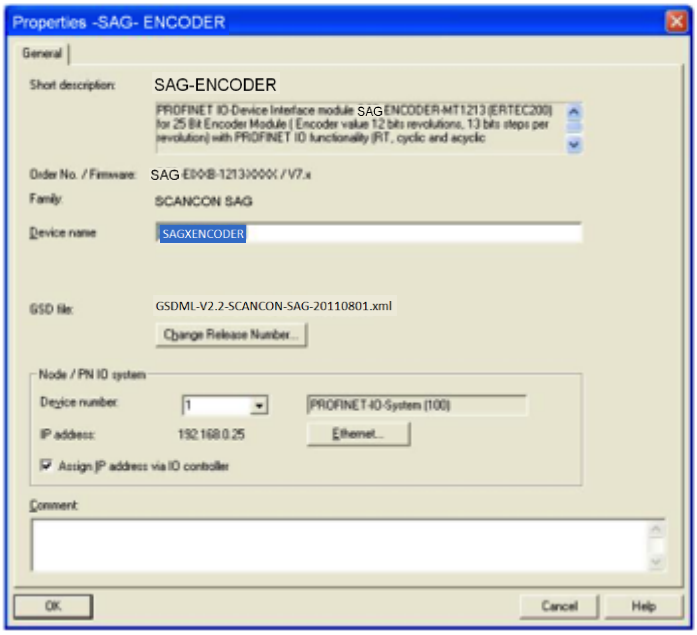


4.2.2 Standard Encoder with PDEV

Asynchronous + RT- + IRT-Communication for Controller which supports IRT functionality (Standard)



Double-click the rotary encoder icon to set communication parameters that the PLC will use. Set a device name and by clicking “Ethernet” the IP address of the SCANCON encoder. Also, under the “IO cycle” tab, set the desired update time.

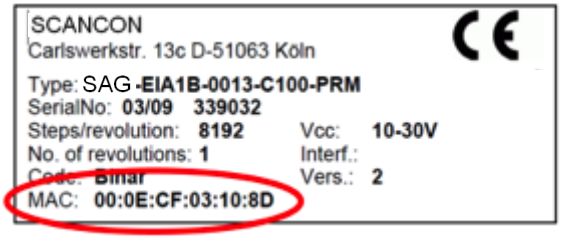


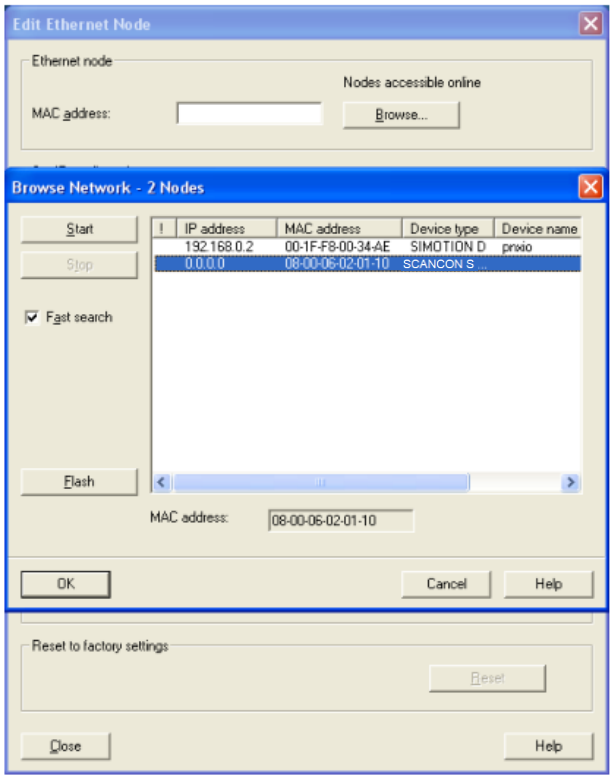
The device name and IP address now have to be set physically within the rotary encoder. Connect the PLC and rotary encoder to ethernet and switch them on. Click "PLC" -> "Ethernet" -> "Edit Ethernet Node" and click "Browse" for accessible ethernet nodes in the new window. STEP7 will scan for devices on Ethernet and will displays them in a window. The rotary encoder should be displayed under the device type "SCANCON SAG". Select this entry and click "Flash" to have the identification LED flash with 2 Hz. Click "OK" to take the MAC address of the chosen device to the previous window and select "Use IP parameters".

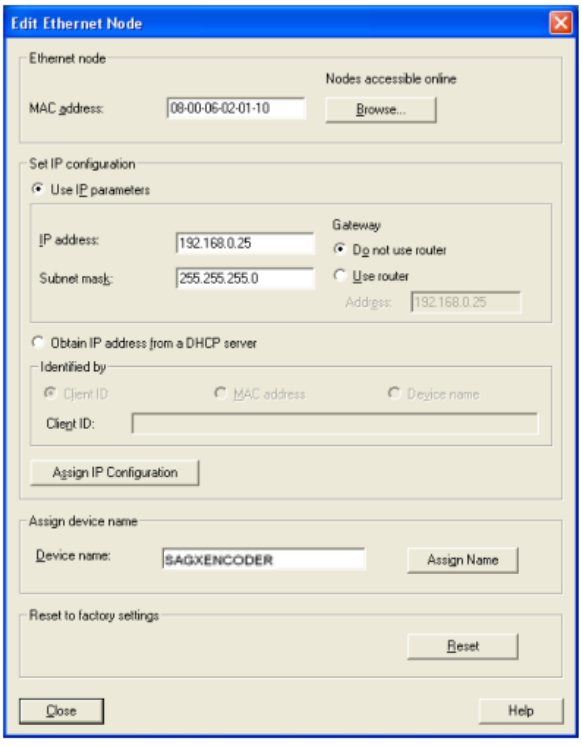
The MAC address is available on the type label on the bottom left (see picture below with red marking). Enter the IP address (and subnet mask) for the encoder that you previously assigned and click "Assign IP configuration". Also, enter the device name previously chosen in the text field "Device name" and click "Assign Name".

Please note:

If more than one rotary encoder is used in the same PROFINET network, each encoder must have a different name and each encoder must be assigned its name before another is connected to the network.

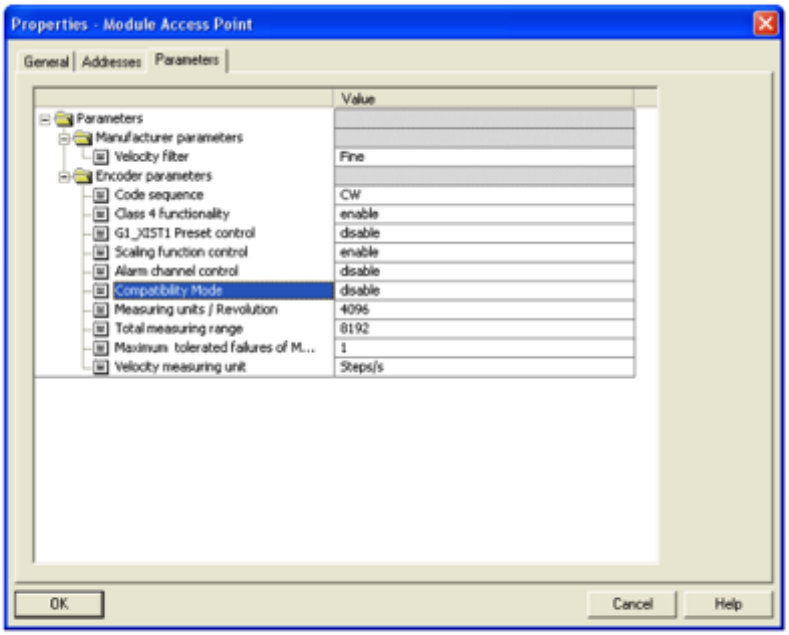






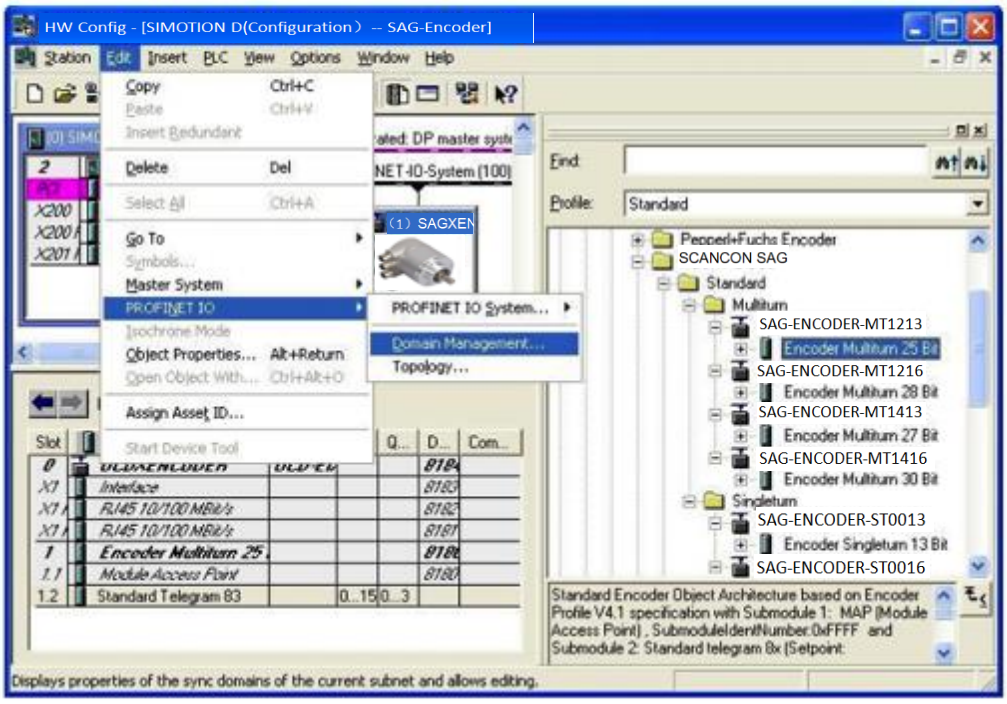
4.3 Module Access Point Parameter setup :

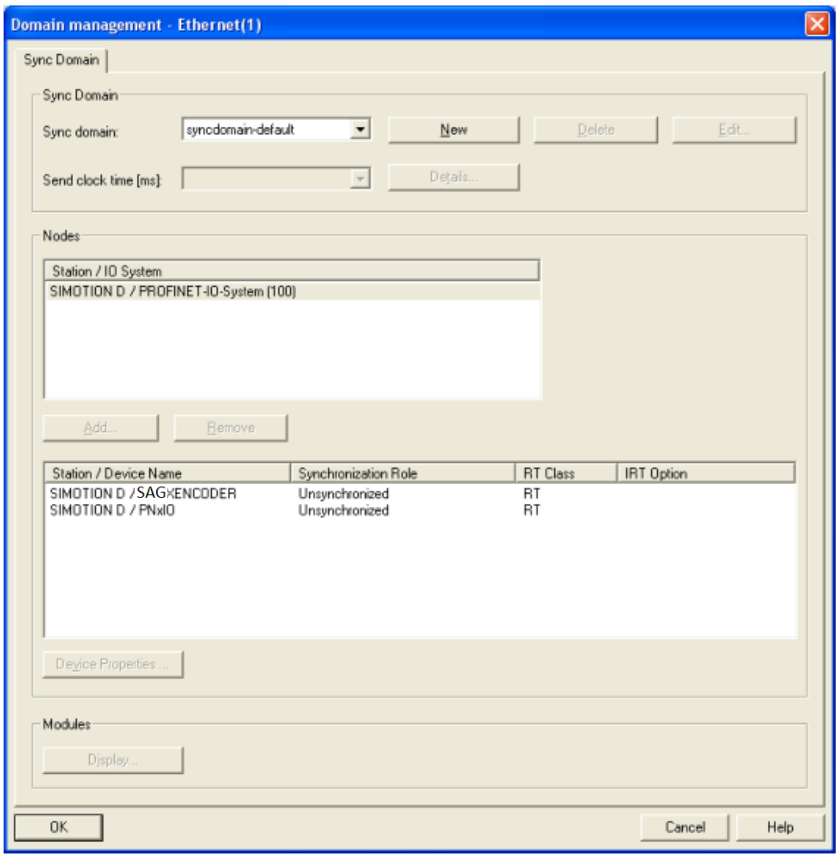
A double click on the Module Access Point will open the window with the list of paramters. This paramters will transmit to the encoder on each start of the PLC.



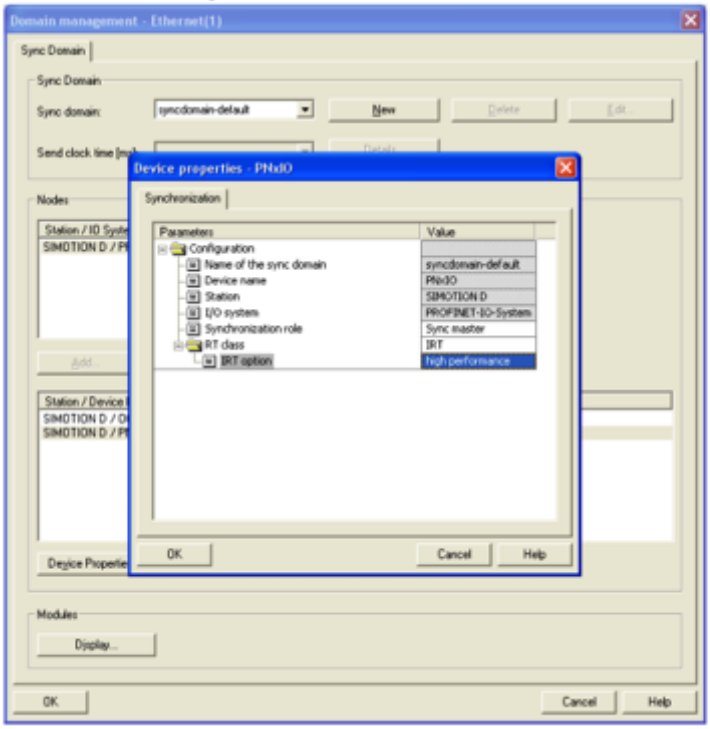
4.4 HW Config IRT-Setup:

On the next screenshots are the necessary steps available for an IRT coummunication.

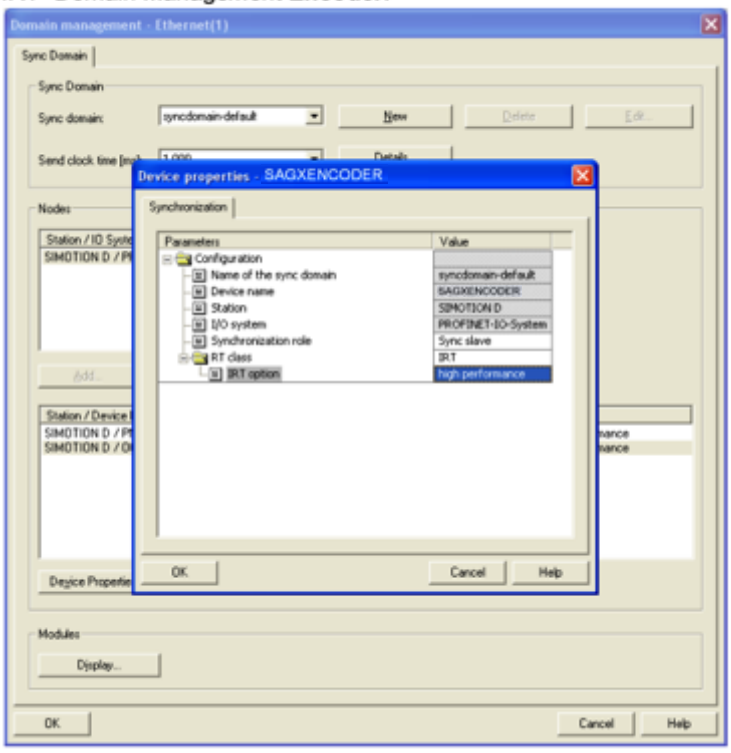




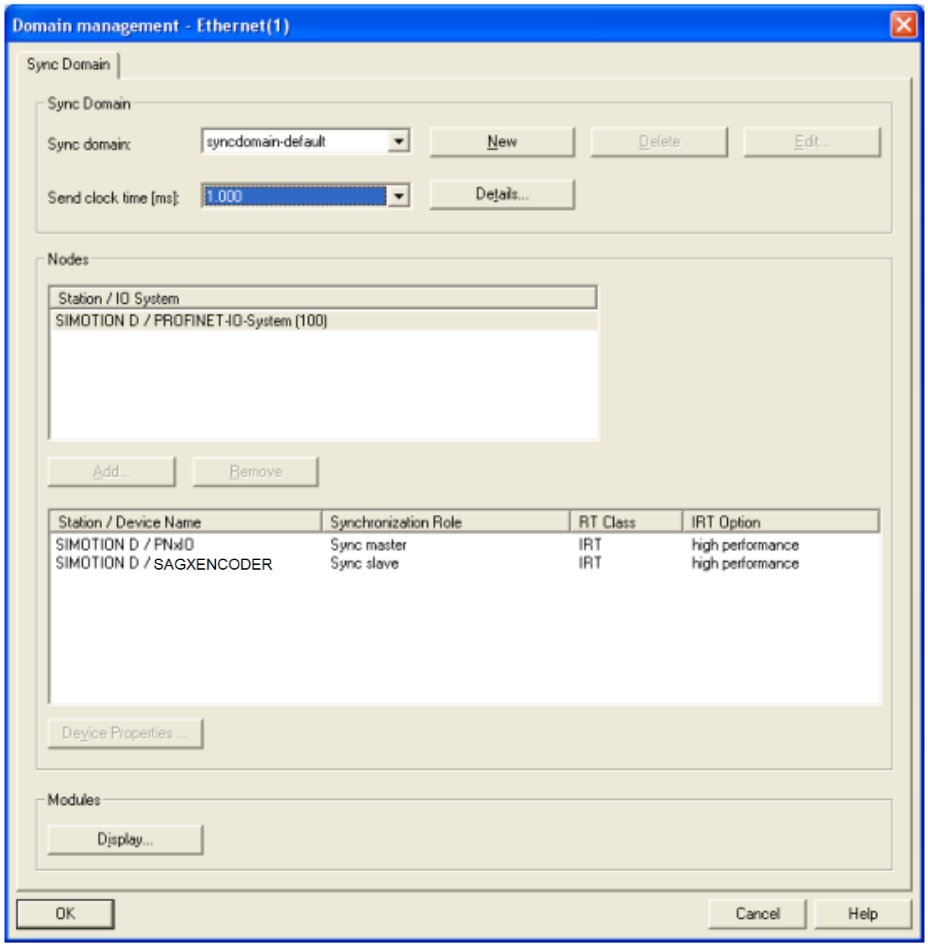
IRT- Domain Management **Controller**



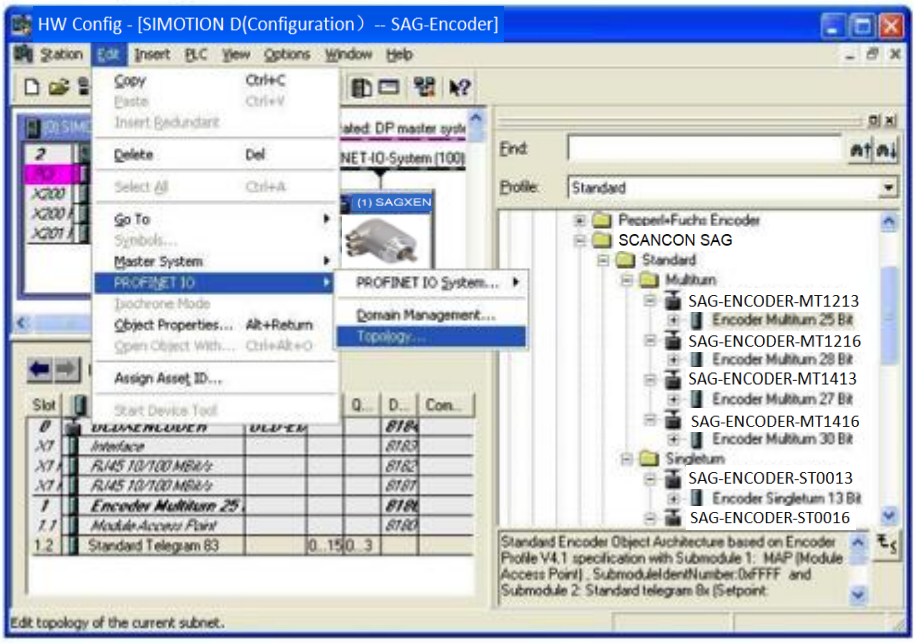
IRT- Domain Management **Encoder**:



Sync Clock:


Domain management - Ethernet(1)
Sync Domain
Sync Domain
Sync domain:
syncdomain-default
New
Delete
Edit...
Send clock time [ms]:
1.000
Details...
Nodes
Station / IO System
SIMOTION D / PROFINET-IO-System (100)
Add...
Remove
Station / Device Name
Synchronization Role
RT Class
IRT Option
SIMOTION D / PNxIO
Sync master
IRT
high performance
SIMOTION D / SAGXENCODER
Sync slave
IRT
high performance
Device Properties ...
Modules
Display...
OK
Cancel
Help

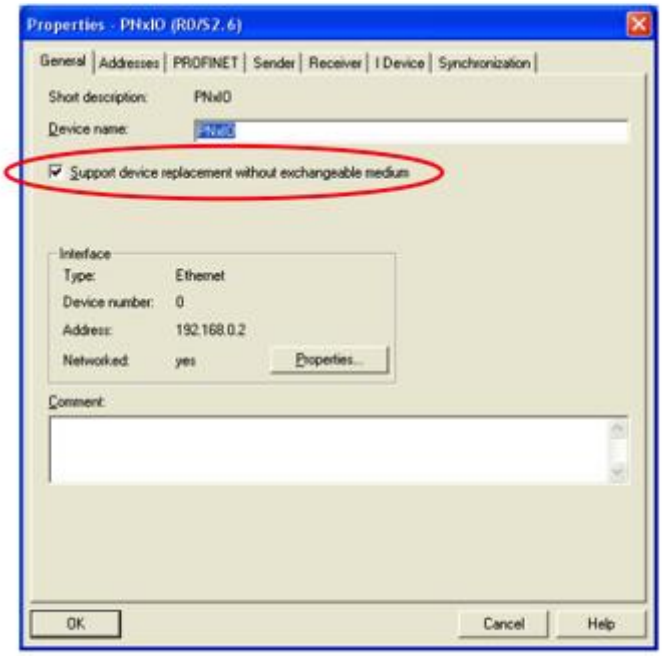
4.5 IRT- Topology...



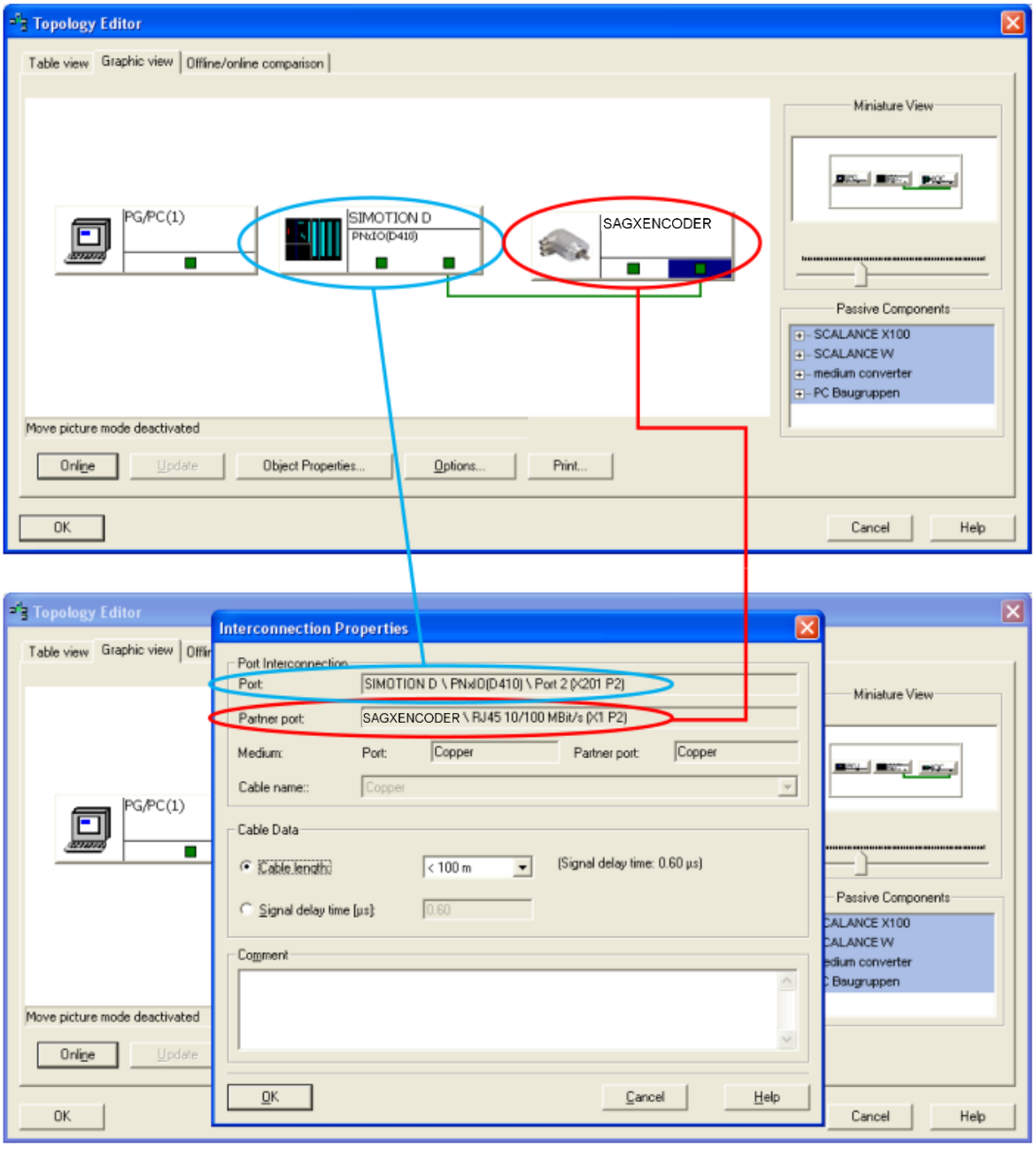
4.6 LLDP (Link Layer Discovery Protocol)

The Link Layer Discovery Protocol allows replacing a device of the Profinet-network. The partner port before and behind of the replaced device save relevant information's so that no additional configuration is necessary.

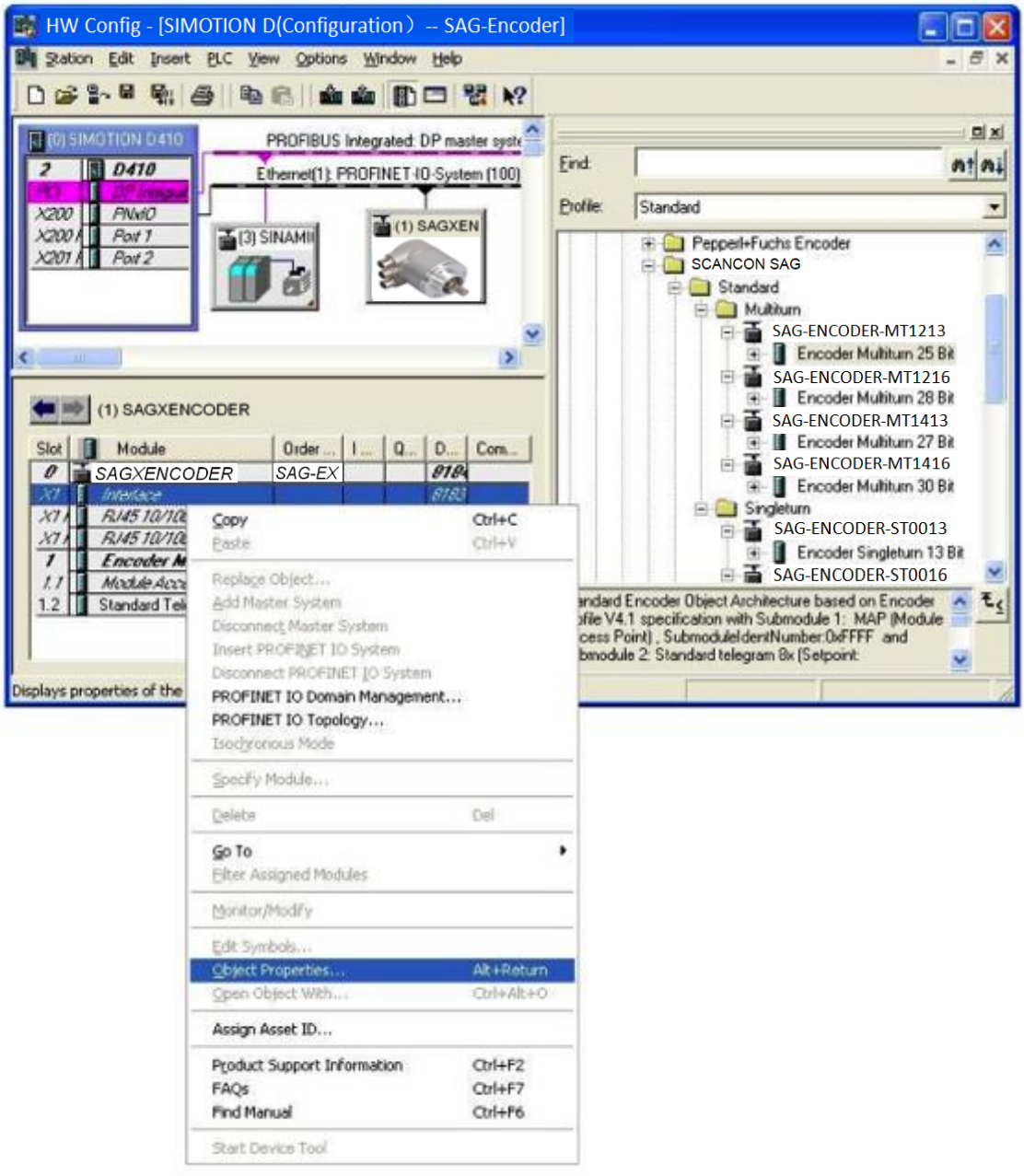
But the flag for activate **"Support Device replacement without replacement medium"** must be activated in Object Properties Interface under tab General.



Topology-setup:

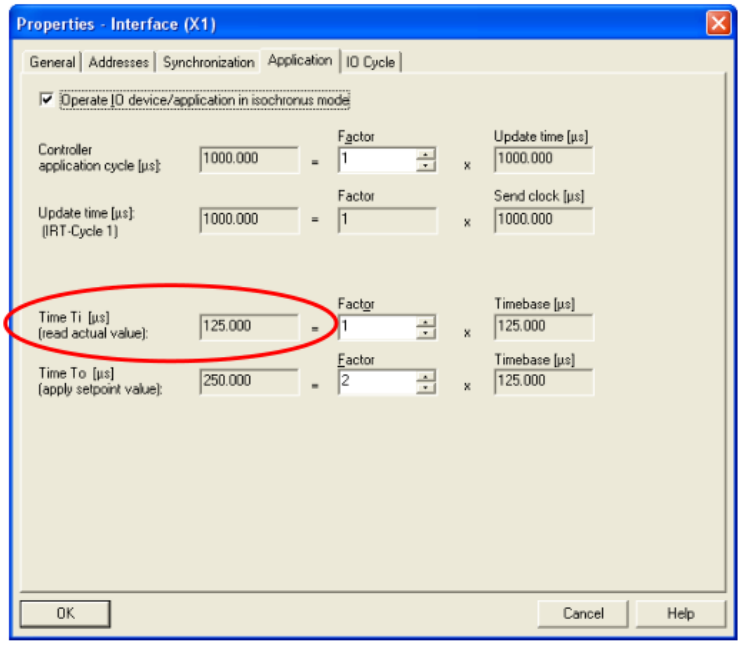


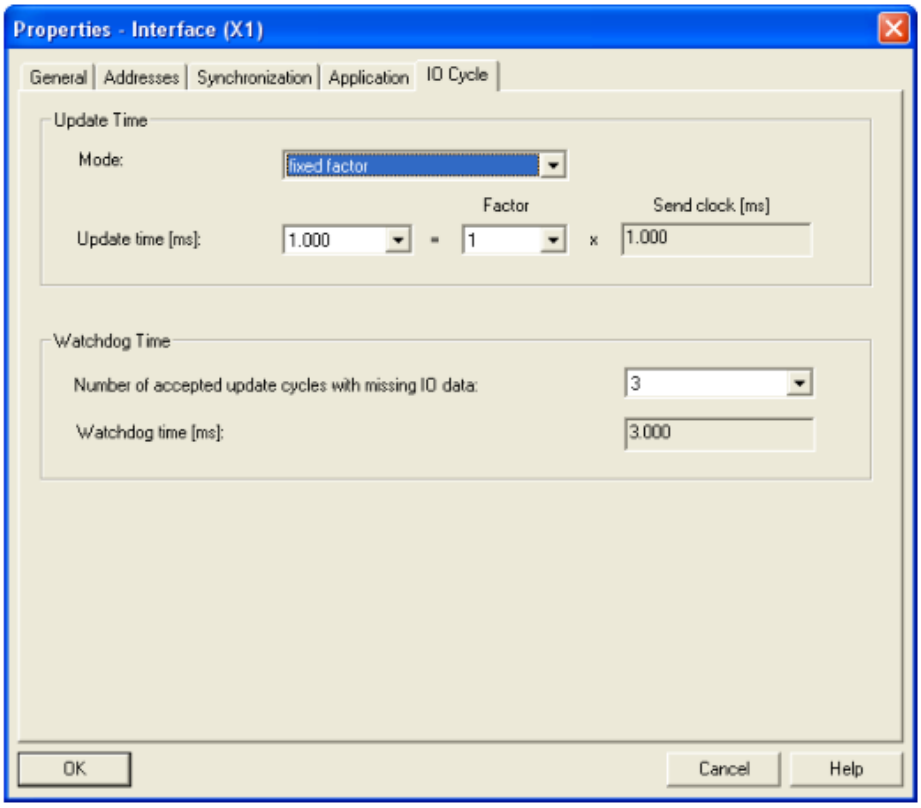
IRT- Encoder Interface X1 Dialog:



IRT- Encoder interface X1 Tab Application:

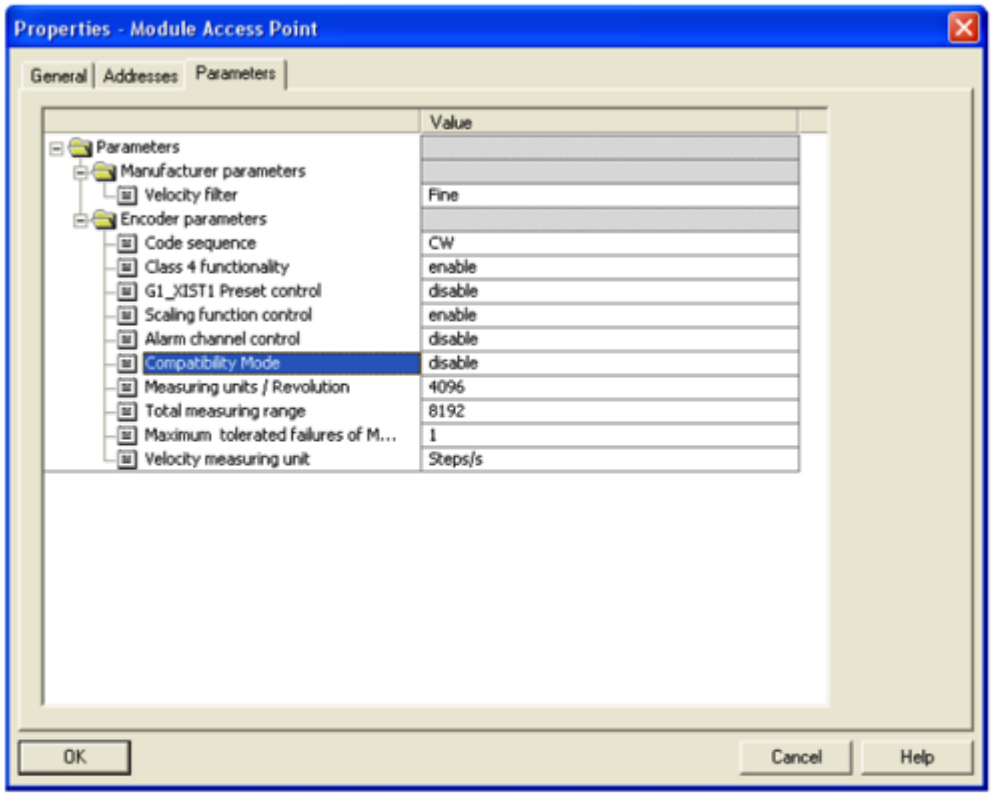
The minimum time for **Ti** is 125µs.





IRT- Sign-Of-Life in Dialog Module Access Point Slot 1 Subslot 1:

Only for IRT-top (High Performance) Synchronous Application



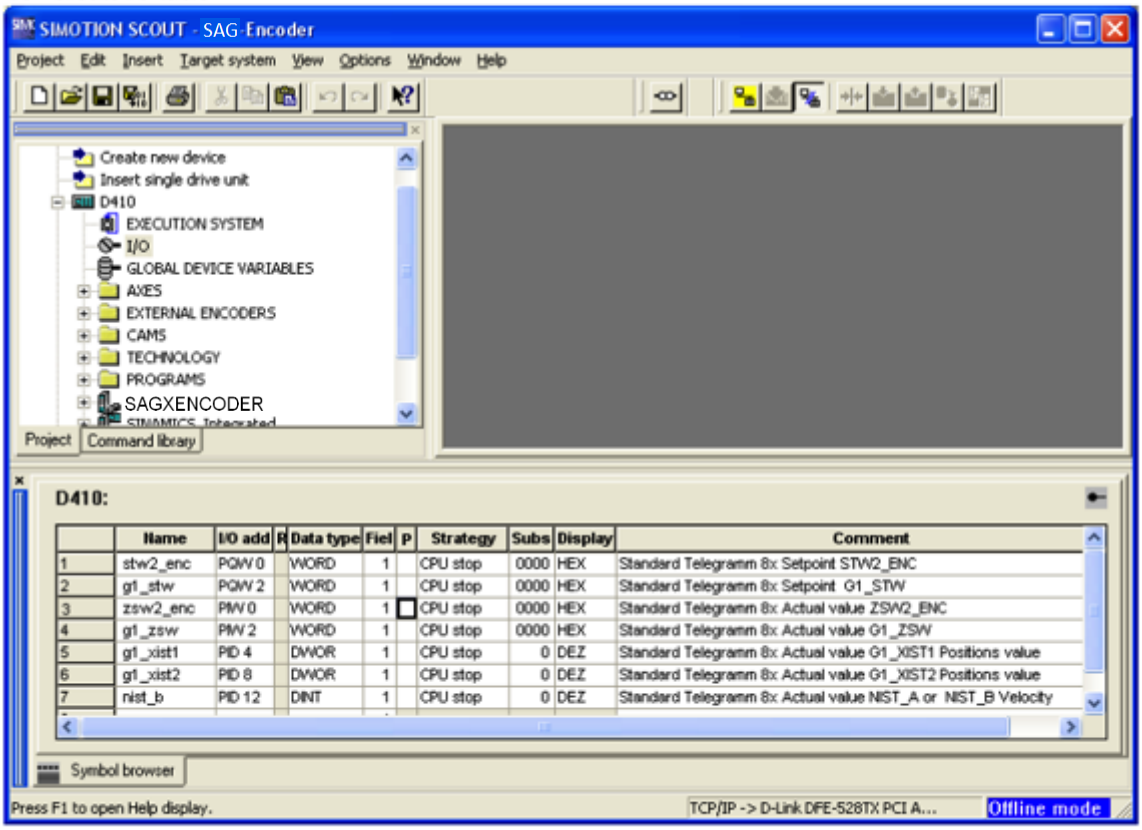
Controller Life Sign Monitoring:

- IRT- Sign-Of-Life Monitoring *active*:
  Compatibility mode disable

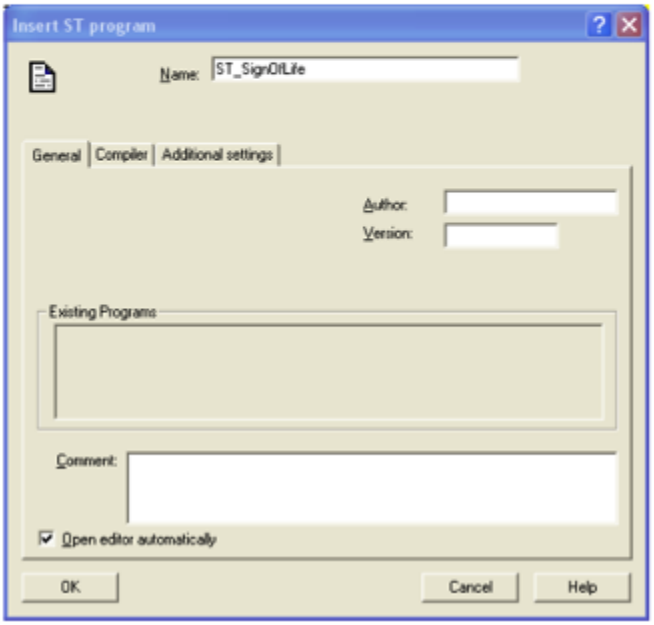
- IRT- Sign-Of-Life Monitoring *not active*:
  Compatibility mode enable

4.7 SIMOTION SCOUT

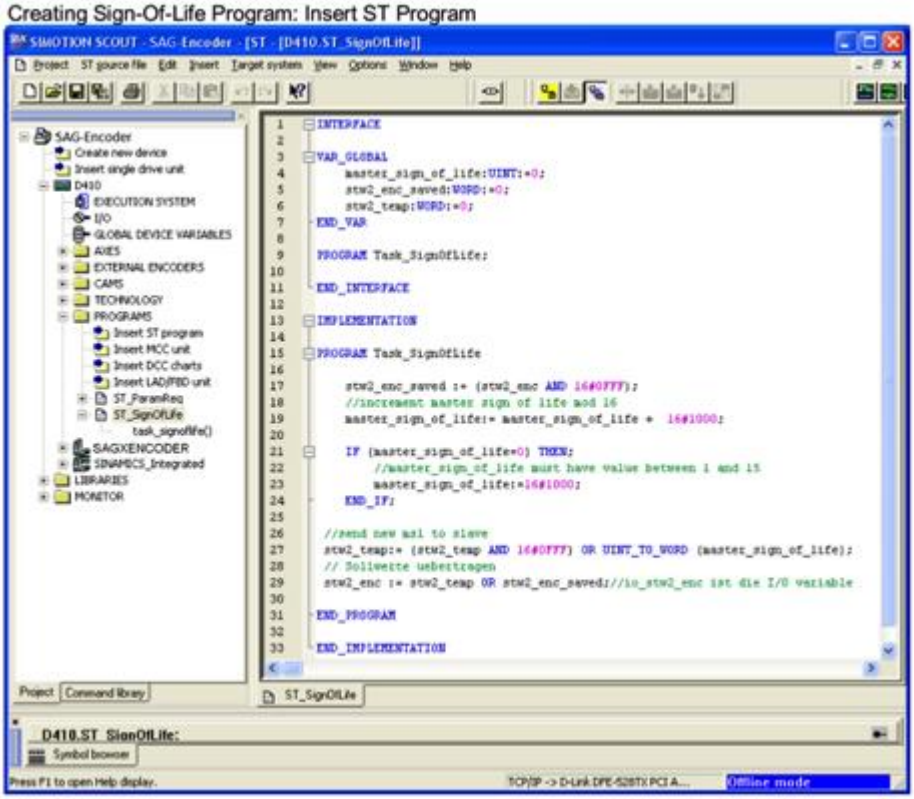
IO- Create IO table



SIMOTION SCOUT IRT-Top Setup: Sign-Of-Life Monitoring for Motion synchronous Application



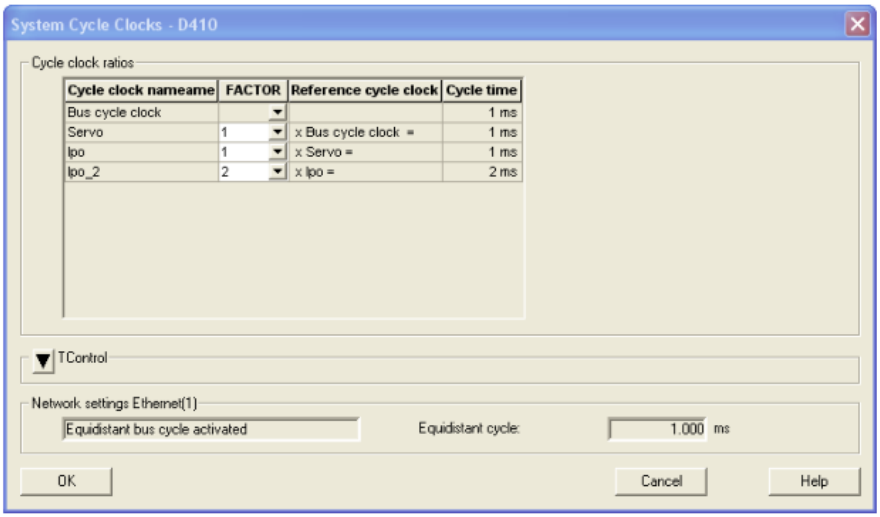
Creating Sign-Of-Life Program: Insert ST Program



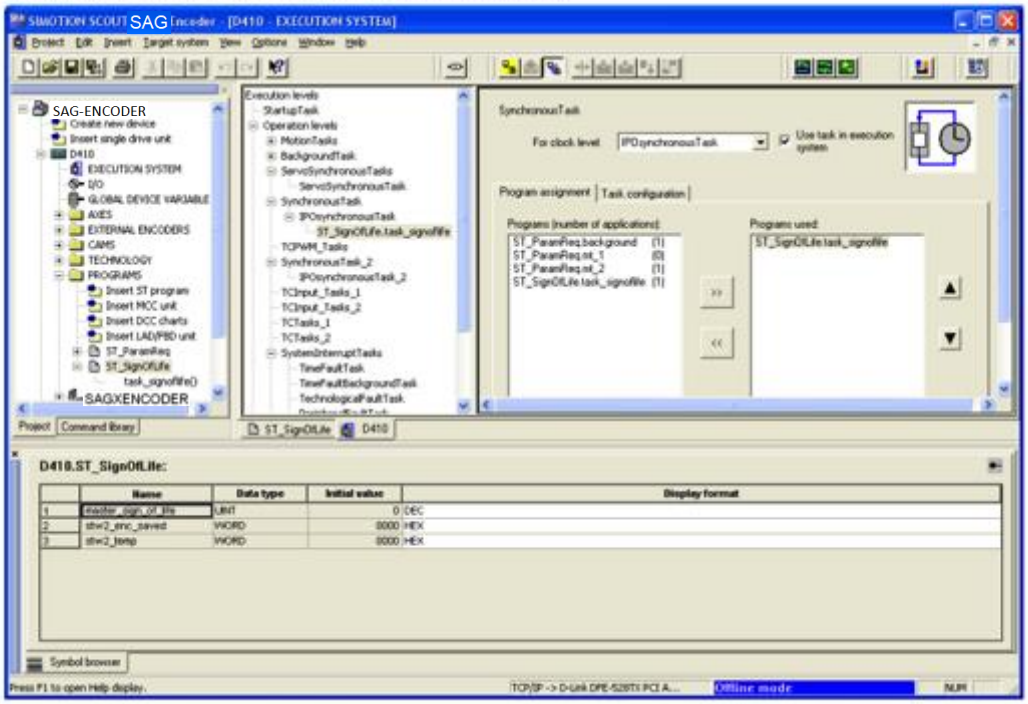
ST-SignOfLife Code edit

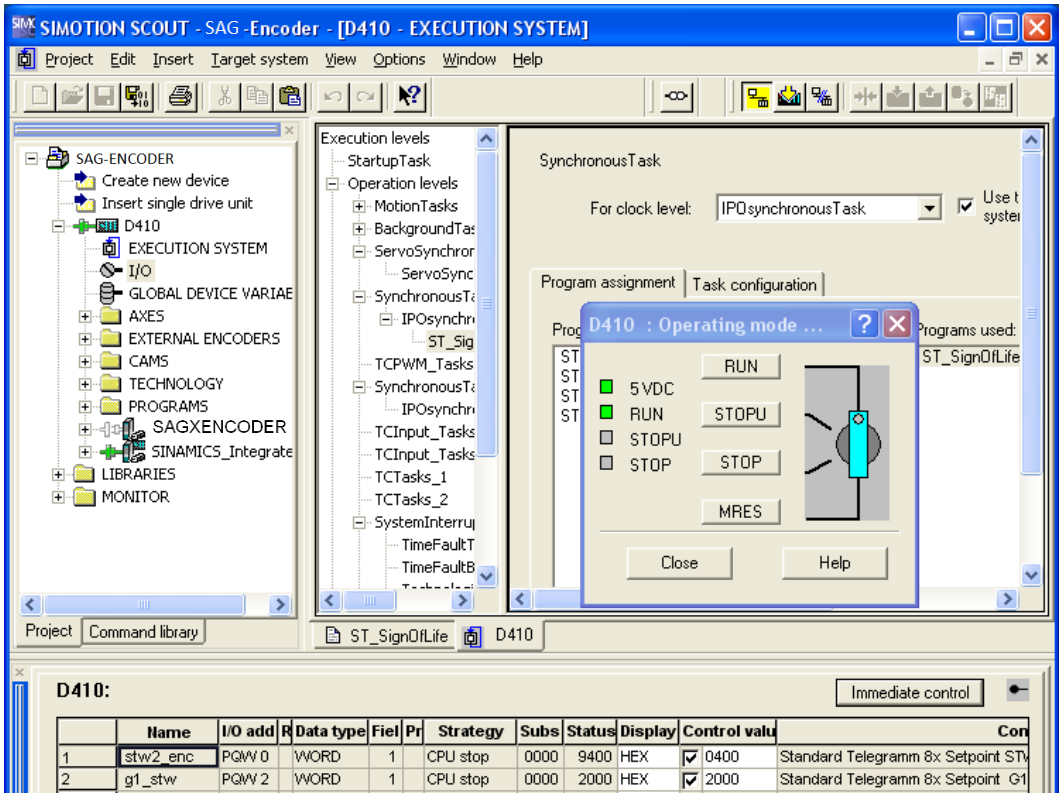
```
INTERFACE

VAR_GLOBAL
    master_sign_of_life:UINT:=0;
    stw2_enc_saved:WORD:=0;
    stw2_temp:WORD:=0;
END_VAR

PROGRAM Task_SignOfLife;

END_INTERFACE

IMPLEMENTATION

PROGRAM Task_SignOfLife

    stw2_enc_saved := (stw2_enc AND 16#0FFF);
    //increment master sign of life mod 16
    master_sign_of_life:= master_sign_of_life +  16#1000;

    IF (master_sign_of_life=0) THEN;
        //master_sign_of_life must have value between 1 and 15
        master_sign_of_life:=16#1000;
    END_IF;

 //send new msl to slave
 stw2_temp:= (stw2_temp AND 16#0FFF) OR UINT_TO_WORD
(master_sign_of_life);
 // Sollwerte uebertragen
 stw2_enc := stw2_temp OR stw2_enc_saved;
//io_stw2_enc ist die I/O variable

END_PROGRAM

END_IMPLEMENTATION
```

Simotion System clock setup:

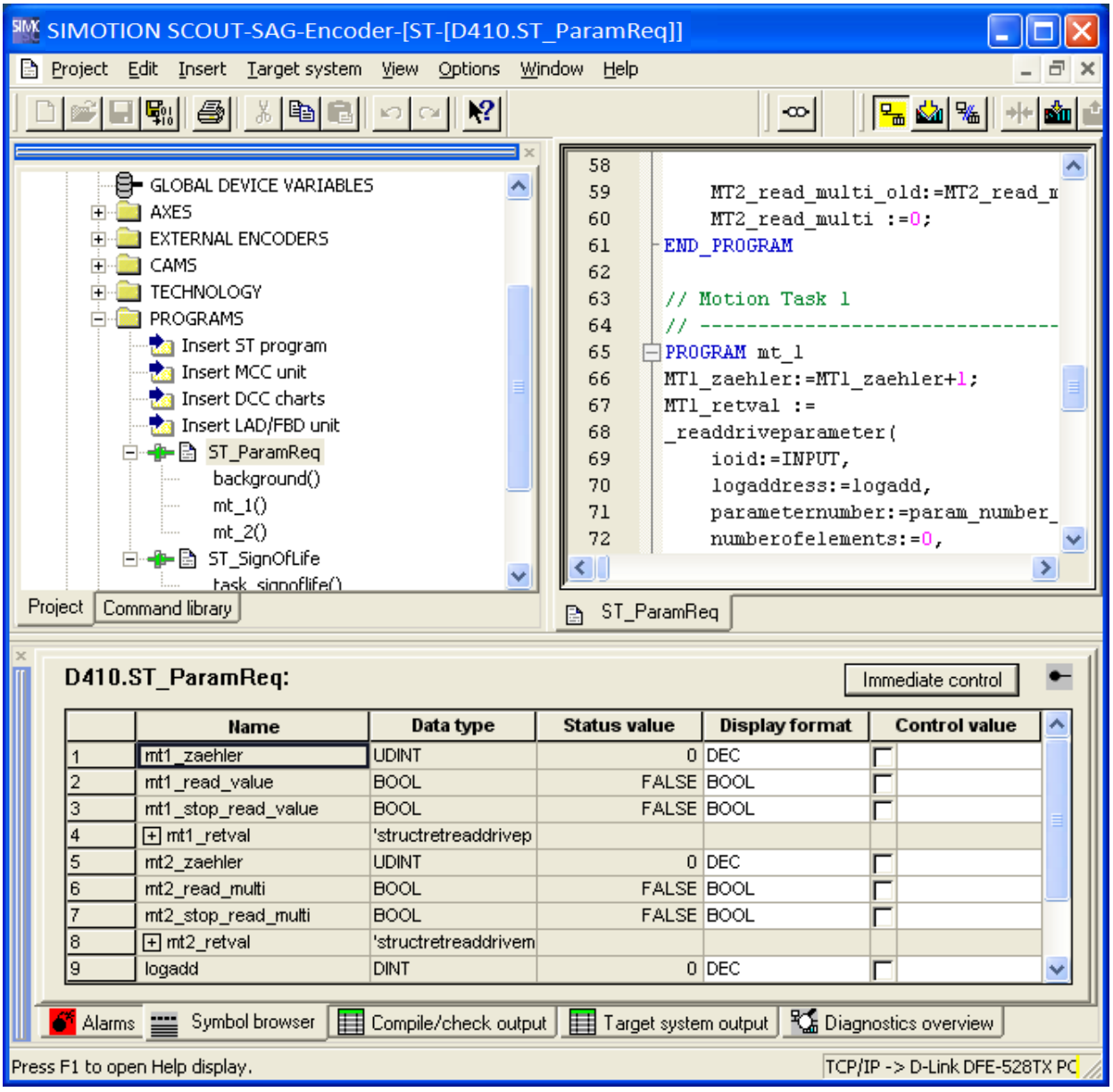


Append SignOfLife-Program on IPOSynchronous task:





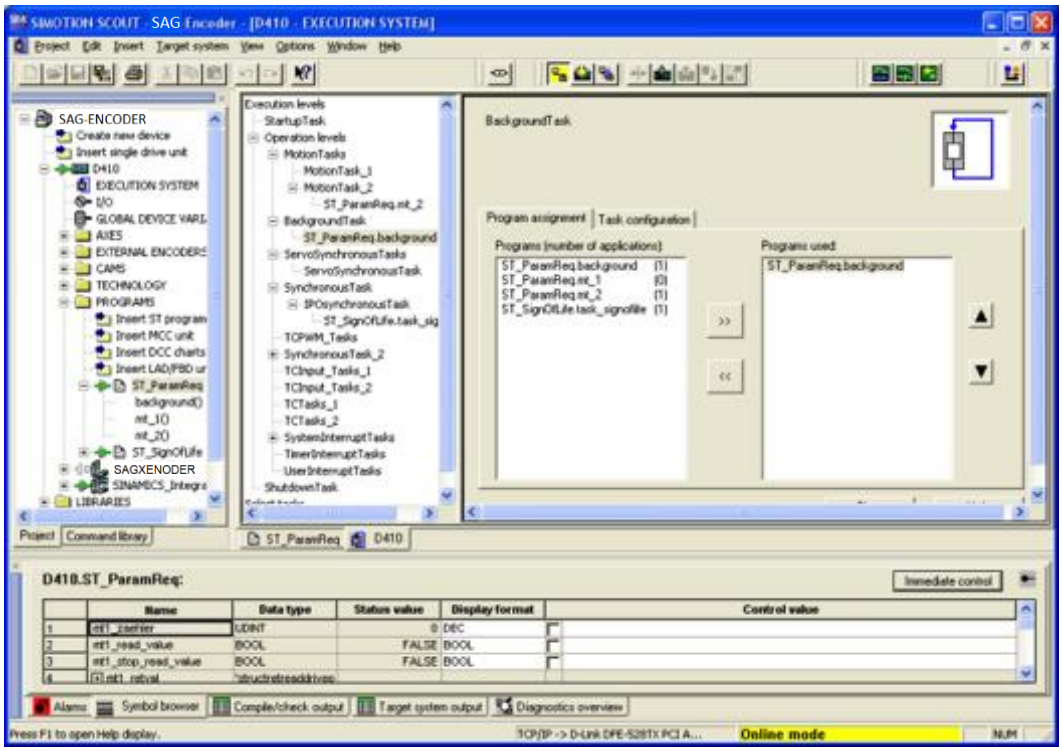
New ST_Program:



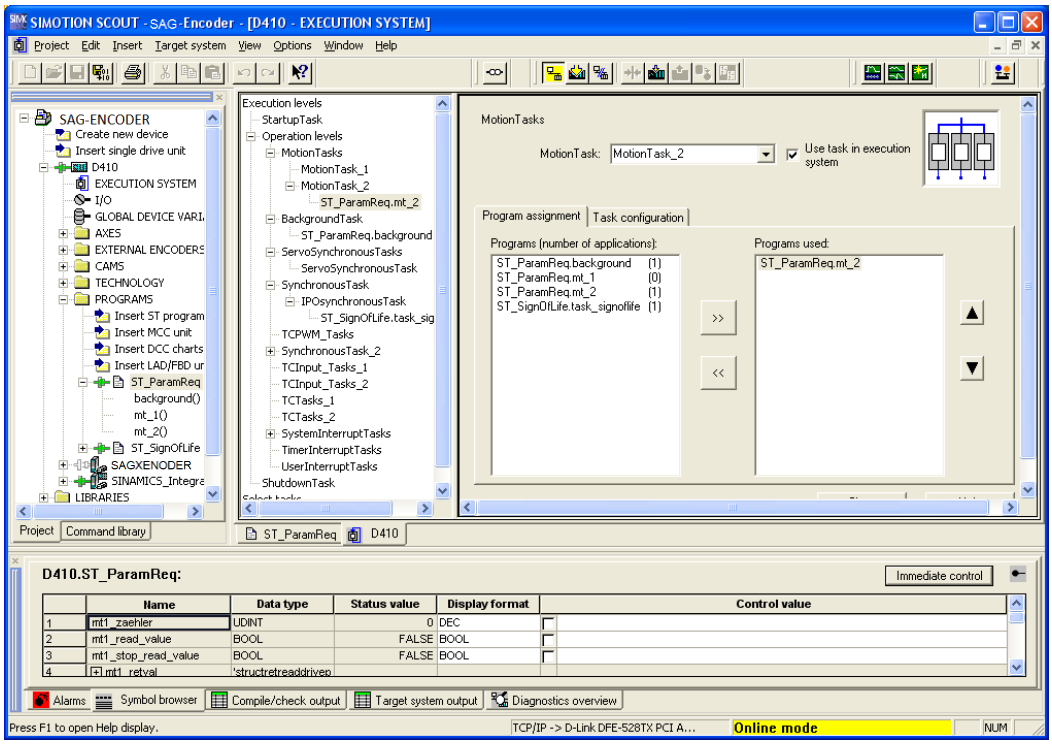
```
// PROGRAM mt_1 Read Single parameter
// PROGRAM mt_2 Read Multi parameter

INTERFACE
  PROGRAM background;
  PROGRAM mt_1;
  PROGRAM mt_2;

 END_INTERFACE

IMPLEMENTATION

VAR_GLOBAL
   //Parameter lesen
   MT1_zaehler:UDINT:=0;
   MT1_read_value:BOOL:=0;
   MT1_stop_read_value:BOOL:=0;
   MT1_retval:StructRetReadDriveParameter;

   MT2_zaehler:UDINT:=0;
   MT2_read_multi:BOOL:=0;
   MT2_stop_read_multi:BOOL:=0;
   MT2_retval:StructRetReadDriveMultiParameter;

   //Parameteraufträge allgemein
   logadd:DINT:=0;
   param_number_single:UDINT:=0;
   param_number_multi:ARRAY [0..38] OF UDINT;
   number_of_param:UDINT:=3;

END_VAR


PROGRAM Background

   VAR
      MT1_read_value_old:BOOL:=0;
      MT2_read_multi_old:BOOL:=0;
   END_VAR

   IF (MT1_read_value=1 AND MT1_read_value_old=0) THEN
      MT1_stop_read_value:=0;
      _starttask(MotionTask_1);
   END_IF;

   MT1_read_value_old:=MT1_read_value;

   number_of_param:=3;
   param_number_multi[0]:=927;
   param_number_multi[1]:=65000;
   param_number_multi[2]:=971;

   IF (MT2_read_multi=1 AND MT2_read_multi_old=0) THEN
      MT2_stop_read_multi:=0;
      _starttask(MotionTask_2);
```

```
    END_IF;

    MT2_read_multi_old:=MT2_read_multi;
    MT2_read_multi :=0;
END_PROGRAM

// Motion Task 1
// ------------------------------------------------------------
PROGRAM mt_1
MT1_zaehler:=MT1_zaehler+1;
MT1_retval :=
_readdriveparameter(
    ioid:=INPUT,
    logaddress:=logadd,
    parameternumber:=param_number_single,
    numberofelements:=0,
    subindex:=0,
    nextcommand:=WHEN_COMMAND_DONE,
    commandid:=(_getCommandID())
);
IF MT1_stop_read_value=0 THEN
    _restarttask(MotionTask_1);
ELSE
    MT1_read_value:=0;
END_IF;
END_PROGRAM


// Motion Task 2 ------------------------------------------------------------
PROGRAM mt_2
MT2_zaehler:=MT2_zaehler+1;

MT2_retval :=
_readdrivemultiparameter(
    ioid:=INPUT,
    logaddress:=logadd,
    numberofparameters:=number_of_param,
    parameternumber:=param_number_multi,
    nextcommand:=WHEN_COMMAND_DONE,
    commandid:=(_getCommandID())
);

//MT2_read_multi:=0;
IF MT2_stop_read_multi=0 THEN
    _restarttask(MotionTask_2);
ELSE
    MT2_read_multi:=0;
END_IF;
END_PROGRAM

END_IMPLEMENTATION
```

Background task:



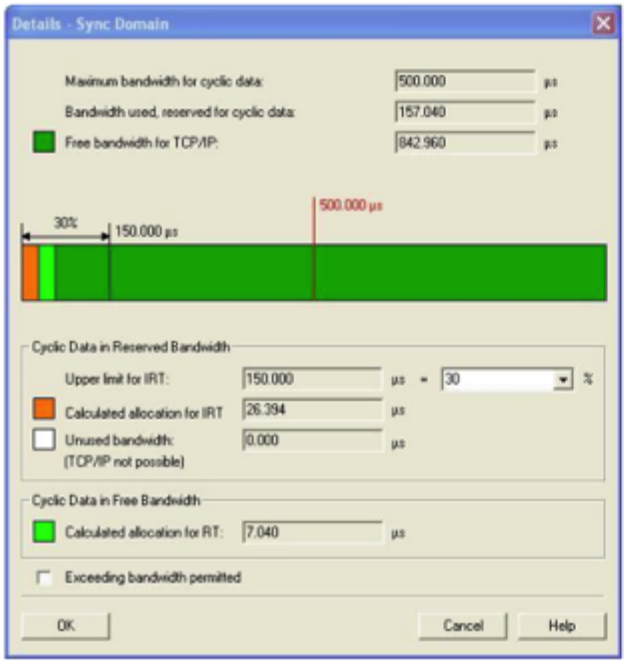
Motiontask_2:



5 IRT communication

5.1 IRT settings

It is possible to set the upper limit for IRT transmission. The smallest time



5.2 User data reliability

5.2.1 General

For both transmission directions (Controller <-> DO), user data reliability is achieved using a Sign-Of-Life (4-bit counter).

The value range of the Sign-Of-Life is only 1 to 15 respectively (0 = invalid) since:

A DO that does not support the fail-safe mode receives a data telegram in the clear mode with the Output Data set to "0" (thus, failure of the Sign-Of-Life may be recognized only if LS = 0 is not permissible).

Through the DO's Sign-Of-Life, a maximum ratio of TMAPC/TDP of 14/1 is possible. Regardless of the ratio TMAPC/TDP, the counter is always incremented to the maximum value (15). In Multi-Axis Drive Units, the reaction to Sign-Of-Life failures is axial. Depending on the device, the reaction to one Drive Axis may affect more Drive Axis.

5.2.2 Controller's Sign-Of-Life (C-LS) Transmission (C-LS)

A 4-bit counter is used in Control Word 2 (refer to 3.4.3) as the Sign-Of-Life for the controller. This counter is incremented by the controller in each controller application cycle, and thus also identifies the computation of the position controller (first DP cycle in the TMAPC). The DO receives the new Sign-Of-Life of the controller together with the new setpoint at the time TO in the following DP-cycle.

Synchronization (C-LS)

The Controller application starts the Controller-LS with an arbitrary value between 1 and 15, at the earliest when changing from Preparation -> Synchronization.

Monitoring (C-LS)

If, in a Controller application cycle, the DO application does not recognize a correct count (i.e. a positive or a negative deviation is recognized), it initially processes with the old telegram data from the last valid controller telegram. For setpoint generation, a device-specific failure strategy may be used.

If the DO application does not recognize the expected numerical value after a parameterized number of controller application cycles (TMLS = n × TMAPC; n may be selected via profile parameter 925; also refer to chapter 5.1.4), the affected Drive Axis messages a fault. After fault acknowledgement, the DO application then attempts to automatically resynchronize itself to the Sign-Of-Life of the controller application. Depending on the particular application, a new start may be required.

If the Sign-Of-Life fails, it may be for the following reasons:

- Sign-Of-Life failure
- Failure of the controller application level (with DP transmission still operational)
- PLL failure
- The DP cycle TDP has been exceeded (through telegram repetition)

Example: Permanent LS failure (see Figure 1), TMLS = 5 × TMAPC: the strategy of the Sign-Of-Life failure counter is explained in chapter 5.1.4:

T_{MAPC}	|	|	|	|	|	|	|	|	|	|
Controller LS (reference):	1	2	3	4	5	6	7	8	9	10
Controller LS (actual):	1	2	**2**	**2**	**2**	**2**	**2**	**2**	**2**	**2**
Failurer counter:	0	0	10	20	30	40	50	50	50	50
Response:			**-> Failure**				-> Switch-off			

Figure 1 – Example: Long term Sign-Of-Life failure of the controller

Example: Temporary LS failure (see Figure 2 and Figure 3), TMLS = 5 × TMAPC: The strategy of the Sign-Of-Life failure counter is explained in chapter 5.2.4:

T_{MAPC}										
Controller LS (reference):	1	2	3	4	5	6	7	8	9	10
Controller LS (actual):	1	2	**2**	**2**	5	6	7	8	9	10
Failurer counter:	0	0	10	20	19	18	17	16	15	14
Response:			**-> Failure**							

Figure 2 – Example: Temporary failure of the controller LS (negative deviation)

T_{MAPC}										
Controller LS (reference):	1	2	3	4	5	6	7	8	9	10
Controller LS (actual):	1	2	**4**	**5**	5	6	7	8	9	10
Failurer counter:	0	0	10	20	19	18	17	16	15	14
Response:			**-> Failure**							

Figure 3 – Example: Temporary failure of the controller LS (positive deviation; double step)

5.2.3 DO's Sign-Of-Life (DO-LS)

Transmission (DO-LS)

A 4-bit counter in status word 2 is used as a Sign-Of-Life for the DO. The DO increments this counter with each DP cycle.

Synchronization (DO-LS)

The DO application starts the DO's Sign-Of-Life with an arbitrary value between 1 and 15: after successful PLL synchronization and at the change (n -> n + 1) of the controller's Sign-Of-Life.

Monitoring (DO-LS)

If the controller application does not recognize a correct count in a controller application cycle (i.e. a positive or negative deviation has been recognized), it initially uses the old telegram data from the last valid DO telegram. To generate the actual value, a device-specific failure strategy may be implemented.

If the controller application does not recognize the expected numerical value after a parameterized time (TSLS = n × TDP; n may be parameterized or defined depending on the manufacturer of the controller application), the affected Drive Axis is shut down by the controller application (possibly also involved drives), and an appropriate fault is signaled to the user. The controller application then attempts to automatically re-synchronize itself to the Sign-Of-Life of the DO application. Depending on the particular application, a re-start may be required or it may be sufficient to acknowledge the fault.

Example reasons for the Sign-Of-Life to fail may be:

- Sign-Of-Life failure
- Failure of the DO application level (while DP transmission is still functioning)
- PLL failure
- DO failure in the sense of DP (DO does not respond although telegram was repeated)

Example: Permanent LS failure (see Figure 4), TSLS = 5 × TDP: the strategy of the Sign-Of-Life failure is explained in chapter 5.1.4:

Time cycle										
DO LS (reference):	1	2	3	4	5	6	7	8	9	10
DO LS (actual):	1	2	**2**	**2**	**2**	**2**	**2**	**2**	**2**	**2**
Failurer counter:	0	0	10	20	30	40	50	50	50	50
Response:			**-> Failure**				-> Switch-off			

Figure 4 – Example: Permanent failure of the DO LS

Example: Temporary LS failure (see Figure 5 and Figure 6), TSLS = 5 × TDP: the strategy of the Sign-Of-Life failure is explained in chapter 5.1.4:

Time cycle										
DO LS (reference):	1	2	3	4	5	6	7	8	9	10
DO LS (actual):	1	2	**2**	**2**	5	6	7	8	9	10
Failurer counter:	0	0	10	20	19	18	17	16	15	14
Response:			**-> Failure**							

Figure 5 – Example: Temporary failure of the DO LS (negative deviation)

Time cycle										
DO LS (reference):	1	2	3	4	5	6	7	8	9	10
DO LS (actual):	1	2	**4**	**5**	5	6	7	8	9	10
Failurer counter:	0	0	10	20	19	18	17	16	15	14
Response:			**-> Failure**							

Figure 6 – Example: Temporary failure of the DO LS (positive deviation; double step)

5.2.4 Counting strategy for the Sign-Of-Life failure counter

The strategy which is applied in order to prevent fast shutdown for a sporadically faulted controller or DO application is described in the following text. This strategy guarantees that at least a specific percentage of the telegrams shall be valid before a Drive Axis is powered down. A counter is defined on the DO side in which for each deviation (independently of whether it is a positive or negative deviation) between the expected and actually transferred value for the controller Sign-Of-Life, it is incremented by ten. For each additional deviation, the counter is again incremented by ten. If a deviation between the expected and received controller Sign-Of-Life is not recognized, the counter is decreased by one. The minimum value which may then be counted down to is zero. This is simultaneously the value from which counting is started. This method ensures that more than 90 % of the telegrams transferred in continuous operation originate from an undisturbed controller application.

Profile parameter 925 (axis-specific, data type Unsigned16) may be used to set a maximum on how many consecutive controller Sign-Of-Life failures may occur (for an initial counter value of zero and without any intermediate valid sequences) without failure of a Drive Axis.

Depending on the previous history, it is possible that just a few controller Sign-Of-Life failures are sufficient to cause a failure of a Drive Axis. If the Drive Axis is powered-down, the Sign-Of-Life failure counter maintains its value up to the start of the re-synchronization operation.

In the example in Figure 7, the Sign-Of-Life failure counter in the Drive Axis is viewed over time with respect to the transferred controller Sign-Of-Life. The maximum number of controller Sign-Of-Life failures which may be tolerated was set to three in parameter 925.

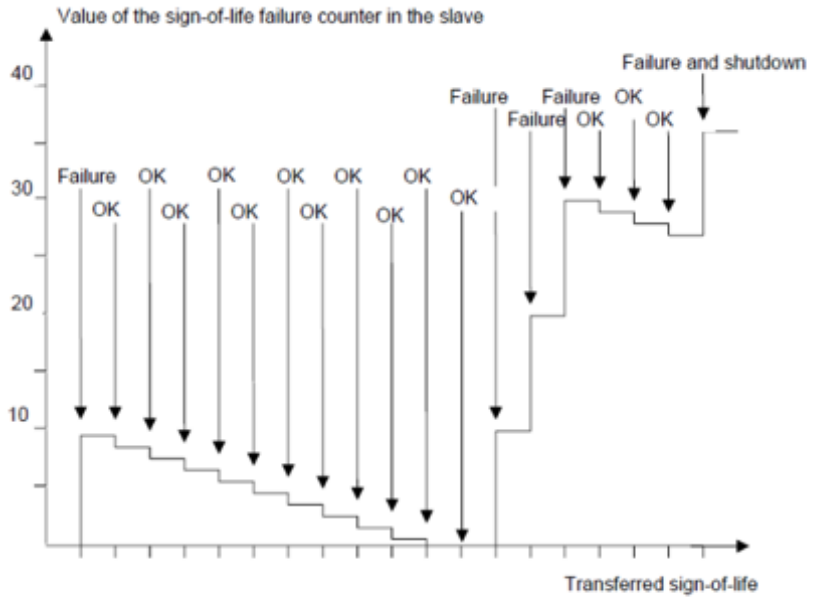


Figure 7 – Value of the DO Sign-Of-Life failure counter (axis-specific) with respect to the transferred controller Sign-Of-Life

The same strategy is recommended when monitoring the DO Sign-Of-Life in the controller. However, it has not been defined with which parameter the maximum number of tolerable DO Sign-Of-Life character failures may be parameterized.

5.2.5 Error codes in G1_XIST2

Error codes are sent in G1_XIST2 if an error occurs.

NOTE! In Clock cycle synchronous applications the encoder additionally indicates the error described by error code 0x0F04 (Synchronization fault) by setting the encoder's Sign-Of-Life to zero (S-LS = 0)

G1_XIST2	Meaning	Explanation
0x0F04	Synchronization fault	The number of permissible failures for the bus cycle signal was exceeded.

5.3 Base Mode Parameter Access

5.3.1 General

In this subclause, the access to parameters via the "Base Mode" is defined. A request language will be defined for the access. The requests and the replies are transmitted acyclically by use of the "Acyclic Data Exchange" mechanism of the Communication System.

The Base Mode Parameter Access exists because of compatibility reasons due to former PROFIdrive profile and every drive shall be able to handle the Base Mode Parameter Access (mandatory).

5.3.2 General characteristics

- 16-bit wide address each for parameter number and subindex.
- Transmission of complete arrays or parts of them, or the entire parameter description.
- Transmission of different parameters in one access (multi-parameter requests).
- Always just one parameter request is being processed at a time (no pipelining).
- A parameter request/parameter response shall fit in a data block (240 bytes default.) The requests/replies are not split-up over several data blocks. The maximum length of the data blocks may be less than 240 bytes depending on Device characteristics or bus configuration.
- No spontaneous messages will be transmitted.
- For optimized simultaneous access to different parameters (for example, operator interface screen contents), "multi-parameter" requests will be defined.
- There are no cyclic parameter requests.
- After run-up, the profile-specific parameters shall be at least readable in every state.

5.3.3 DO addressing modes

The Base Mode Parameter Access is defined with two different DO address modes according to the following definition:

- Base Mode Parameter Access – Local: In this address mode, only the local parameters of the DO are accessible, to which the CO, where the parameter access point is attached, is related. Access of all global parameters is also possible. The DO-ID in the parameter request header is of no significance.
- Base Mode Parameter Access – Global: In this address mode, all parameters of the Drive Unit are accessible, to which the CO, where the parameter access point is attached, is related. The DO-ID in the parameter request is used for accessing of local parameters inside the Drive Unit. For access of global parameters, the DO-ID 0 may also be used. This address mode serves for compatibility reasons (PROFIBUS) and should not be used by new PROFINET IO controller and Supervisor application processes.

5.3.4 Parameter requests and parameter responses

A parameter request consists of three segments:

Request header

ID for the request and number of parameters which are accessed. Multi-Axis and Modular drives, Addressing of one DO.

Parameter address

Addressing of a parameter. If several parameters are accessed, there are correspondingly many parameter addresses. The parameter address appears only in the request, not in the response.

Parameter value

Per addressed parameter, there is a segment for the parameter values. Depending on the request ID, parameter values appear only in either the request or in the reply.

The following telegram contents are displayed in words (a word or 2 bytes per line). Words or double words will have the most significant byte being transmitted first (big endian) (see Figure 8).

Words and double words:

Word:	Byte 1	Byte 2

Double word:	Byte 1	Byte 2
	Byte 3	Byte 4

Figure 8 – Byte order for Words and Double words

According to the Base Mode Parameter Access, the structure of the parameter request and parameter response is shown in the next tables.

Base mode parameter request:

Block definition	Byte n	Byte n+1	n
Request Header	Request Reference	Request ID	0
	Axis-No./DO-ID	No. of Parameters = n	2
1st Parameter Address	Attribute	No. of Elements	4
	Parameter Number (PNU)		
	Subindex		
nth Parameter Address	...		4 + 6 × (n-1)
1st Parameter Value(s)	Format	No. of Values	4 + 6 × n
(only for request	Values		
"Change parameter")	...		
nth Parameter Values	...		
			4 + 6 × n +...+ (Format_n × Qty_n)

Base mode parameter response:

Block definition	Byte n	Byte n+1	n
Response Header	Request Ref. mirrored	Response ID	0
	Axis-No./DO-ID mirrored	No. of Parameters = n	2
1st Parameter Value(s)	Format	No. of Values	4
(only after request	Values or Error Values		
"Request")	...		
nth Parameter Values	...		
			4 +... + (Format_n × Qty_n)

Request Header

- Request Reference Unique identification of the request/response pair for the master. The master changes the request reference with each new request (for example, modulo 255). The slave mirrors the request reference in the response.
- Request ID two IDs are defined:
  - Request parameter
  - Change parameter

  A parameter change may be stored either in volatile or non-volatile RAM according to the device. A changed parameter that is stored in volatile RAM may first be stored in ROM with parameter P971. The differentiation Value/Description/Text is added to the address as an attribute. The differentiation Word/Double Word is added to the parameter values as a format. For the differentiation Single/Array Parameter, refer to "No. of Elements" in the parameter address.
- Response ID

  Mirroring of the request ID with supplement information whether the request was executed positively or negatively.

  – Request parameter positive

  – Request parameter negative (it was not possible to execute the request, entirely or partially)

  – Change parameter positive

  – Change parameter negative (it was not possible to execute the request, entirely or partially)

  If the response is negative, error numbers are entered per partial response instead of values.
- Axis-No./DO-ID For Base Mode Parameter Access – Local: irrelevant; In the parameter response, the DOID out of the request is mirrored.

  For Base Mode Parameter Access – Global: DO addressing information used for Multi-Axis or Modular drives. This enables various axes/DOs to be able to be accessed each with a dedicated parameter number space in the drive via the same PAP.
- No. of Parameters

  In the case of multi-parameter requests, specifying the number of the following Parameter

  Address and/or Parameter Value areas. For single requests the No. of parameters = 1. Default value range 1 to 39. The value range may be reduced or extended, which shall be indicated by P974.

  Notice, that for a multi-parameter request the PROFIdrive Drive Unit shall arrange the parameter value areas in the response message in the same order as in the corresponding multi-parameter request message.

Parameter Address

- **Attribute**
  Type of object which is being accessed. Value range:
  – Value
  – Description
  – Text
- **Number of Elements**
  Number of array elements that are accessed or length of string which is accessed.
  Default value range 0, 1 to 234. The value range may be reduced or extended which shall be indicated by P974.
  Special Case Number of Elements = 0:
  If values are accessed: recommended for non-indexed parameters.
- **Parameter Number**
  Addresses the parameter that is being accessed. Value range: 1 to 65535.
- **Subindex**
  Addresses the first array element of the parameter or the beginning of a string access or the text array, or the description element that is being accessed. Value range: 0 to 65 535.

Parameter Value

- **Format**
  Format and number specify the location in the telegram to which subsequent values are assigned.
  Value range:
  – Zero (without values as positive partial response to a change request)
  – Data type
  – Error (as negative partial response)
  – Instead of a data type, the following are possible:
  – Byte (for description and texts)
  – Word
  – Double word
- **Number of Values**
  Number of the following values or number of the following data type elements (number of octets in case of OctetString). In case of write request of OctetString, the correct length shall be supplied otherwise the drive shall responds with error 0x18, "number of values are not consistent" (see Table 32).
- **Values**
  The values of the parameter
  If the values consist of an odd number of bytes, a zero byte is appended in order to secure the word structure of the telegrams.

In the case of a **positive partial response**, the parameter value contains the following:
– Format = (Data Type or Byte, Word, Double Word)
– Number of values
– the values

In the case of a **negative partial response**, the parameter value contains the following:
– Format = error
– No. of values = 1
– Value = error value = error number

In the case of a **negative response**, the parameter value may contain the following:
– Format = error
– No. of values = 2
– Value 1 = Error Value 1: error number
– Value 2 = Error Value 2: subindex of the first array element where the error occurs
– (Purpose: after a faulty write access to an array, not all values shall be repeated)

In the case of a **positive partial response without values**, the parameter value contains the following:
– Format = zero
– Number of values = 0
– (no values)

Not all combinations consisting of attribute, number of elements, and subindex are permitted (refer to next table). A parameter which is not indexed in the profile may be realized with indices in the Drive Unit, if the response to a Parameter Access is profile-specific.

Attribute	No. of Elements	Subindex	Related Data
Value (single parameter)	0	0	The value
	1	0	The value
(Indexed parameter)	1	0 - n	One value, under subindex
	2 - n[a]	0 - n	Several values, starting with subindex
Description	0 (irrelevant)	0	The entire description
	1	1 - n	One description element
Text (from text array)	1	0 - n	One text (16bytes), under subindex
	2 - n	0 - n	Several texts, starting with subindex

[a] If the number of elements available in the device does not match with the number of elements which are requested or shall be changed, an error shall be output.

5.3.5 Coding

The coding of the fields in parameter request / parameter response of Base Mode Parameter Access:

Field	Data Type	Values		Comment
Request Reference	Unsigned8	0x00 0x01 - 0xFF	reserved	
Request ID	Unsigned8	0x00 0x01 0x02 0x03 - 0x3F 0x40 - 0x7F 0x80 - 0xFF	reserved Request parameter Change parameter reserved manufacturer-specific reserved	
Response ID	Unsigned8	0x00 0x01 0x02 0x03 - 0x3F 0x40 - 0x7F 0x80 0x81 0x82 0x83 - 0xBF 0xC0 - 0xFF	reserved Request parameter(+) Change parameter(+) reserved manufacturer-specific reserved Request parameter(-) Change parameter(-) reserved manufacturer-specific	
Axis/DO-ID	Unsigned8	0x00 0x01 - 0xFE 0xFF	Device-Representative DO-ID-Number 1 - 254 reserved	Zero is not a DO but the access to the Drive Unit representative.
No. of Parameters	Unsigned8	0x00 **0x01 - 0x27** 0x28 - 0xFF	reserved **Quantity 1 - 39** reserved	There may be an additional limitation through the communication system (telegram length) or optional scalability
Attribute	Unsigned8	0x00 0x10 0x20	reserved Value Description	The four least significant bits are reserved for (future) expansion of "No. of elements" to 12 bits.

Field	Data Type	Values		Comment
		0x30	Text	
		0x40 - 0x70 0x80 - 0xF0	reserved manufacturer-specific	
No. of Elements	Unsigned8	0x00 0x01 - 0xEA 0xEB - 0xFF	Special Function Quantity 1 to 234 reserved	Limitation through compatibility with PROFIBUS Process data ASE telegram length.
Parameter Number	Unsigned16	0x0000 0x0001 - 0xFFFF	reserved Number 1 to 65 535	
Subindex	Unsigned16	0x0000 - 0xFFFE	Number 0 to 65 534	
Format	Unsigned8	0x00 0x01 - 0x38 0x39 - 0x3F 0x40 0x41 0x42 0x43 0x44 0x45 - 0x70 0x71 – 0x7C 0x7D – 0xFF	reserved Data types reserved Zero Byte Word Double word Error reserved Data types reserved	Every slave shall at least support the data types Byte, Word and Double Word (mandatory). Write requests by the master preferably use the "correct" data types (refer to Clause 5). As substitute, Byte, Word or Double Word are also possible. The master shall be able to interpret all values/data types.
No. of Values	Unsigned8	0x00 - 0xEA 0xEB - 0xFF	Quantity 0 to 234 reserved	Limitation because of 240 Bytes Datablock size (compatibility reasons).
Error Number	Unsigned16	0x0000 - 0x00FF	Error Numbers (see Table 32)	The more significant byte is reserved.

The device shall output an error, if reserved values are accessed. The error numbers in Base Mode parameter responses:

Error No.	Meaning	Used at	Additional Info
0x00	Impermissible parameter number	Access to unavailable parameter	0
0x01	Parameter value cannot be changed	Change access to a parameter value that cannot be changed	Subindex
0x02	Low or high limit exceeded	Change access with value outside the value limits	Subindex
0x03	Faulty subindex	Access to unavailable subindex of array parameter. Shall not be used for non array parameters	Subindex
0x04	No array	Access with subindex to non-indexed parameter	0
0x05	Incorrect data type	Change access with value that does not match the data type of the parameter	0
0x06	Setting not permitted (may only be reset)	Change access with value unequal to 0 where this is not permitted	Subindex
0x07	Description element cannot be changed	Change access to a description element that cannot be changed	Subindex
0x08	reserved	Compatibility reasons	-
0x09	No description data available	Access to unavailable description (parameter value is available)	0

Error No.	Meaning	Used at	Additional Info
0x0A	reserved	Compatibility reasons	-
0x0B	No operation priority	Change access without rights to change parameters	0
0x0C	reserved	Compatibility reasons	-
0x0D	reserved	Compatibility reasons	-
0x0E	reserved	Compatibility reasons	-
0x0F	No text array available	Access to text array that is not available (parameter value is available)	0
0x10	reserved	Compatibility reasons	-
0x11	Request cannot be executed because of operating state	Access is temporarily not possible for reasons that are not specified in detail	0
0x12	reserved	Compatibility reasons	-
0x13	reserved	Compatibility reasons	-
0x14	Value impermissible	Change access with a value that is within the value limits, but is not permissible for other long-term reasons (parameter with defined single values)	Subindex
0x15	Response too long	The length of the current response exceeds the maximum transmittable length	0
0x16	Parameter address impermissible	Illegal value or value which is not supported for the attribute, number of elements, parameter number or subindex or a combination	0
0x17	Illegal format	Write request: Illegal format or format of the parameter data which is not supported	0
0x18	Number of values are not consistent	Write request: Number of the values of the parameter data do not match the number of elements in the parameter address	0
0x19	Axis/DO nonexistent	Access to an Axis/DO which does not exist	0
0x20	Parameter text element cannot be changed	Change access to a parameter text element that cannot be changed	Subindex
0x21	Service not suported	Illegal Request ID (Response ID = 0x80)	
0x22 - 0x64	reserved	-	-
0x65 - 0xFF	Manufacturer-specific	-	-

In general, every PROFIdrive Drive Unit shall support Base Mode parameter read and write requests with the data types, Byte, Word and Double Word (mandatory). If the PROFIdrive Drive Unit also supports additional data types, it shall behave in the following manner:

- In case of a parameter read request, it shall signal the corresponding data type in the read response.
- In case of a parameter write request it shall check the data type and signal an error if parameter types do not match.

If the PROFIdrive Drive Unit does not support additional data types, it shall behave in the following manner:

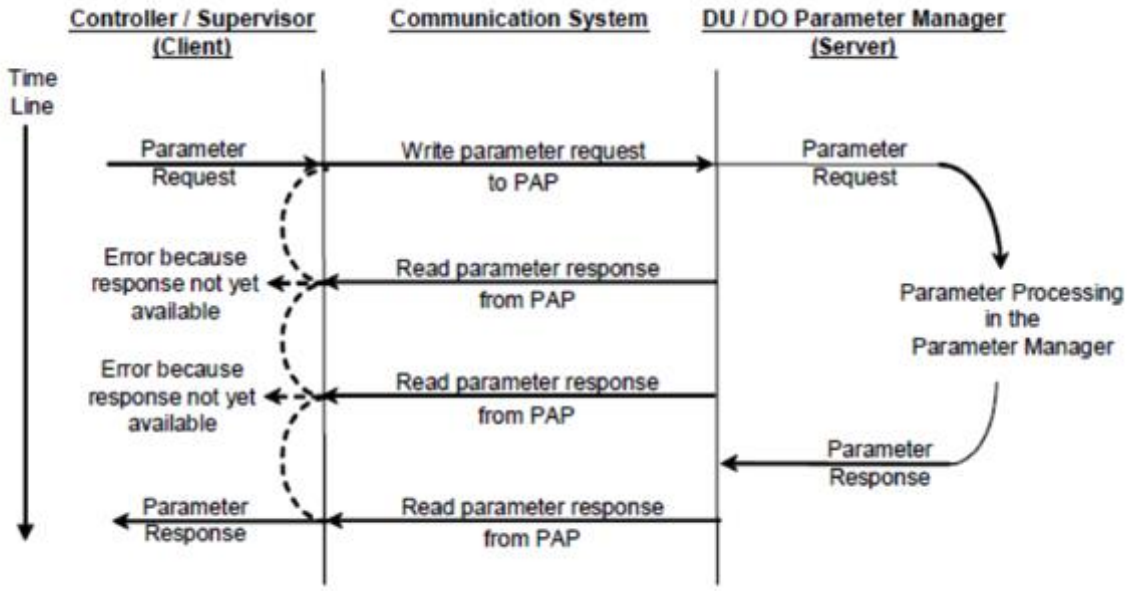
- It rejects the parameter write request with an error response if data types do not match.

The error numbers 0x00 - 0x13 are taken from PROFIdrive Profile, Version 2. Values that cannot be assigned are reserved for future use. If an error with error number 0x05, 0x16, 0x17 or 0x18 occurs while processing a multi parameter change value request, all further parameter requests in the multi parameter request shall be aborted.

5.3.6 Data flow

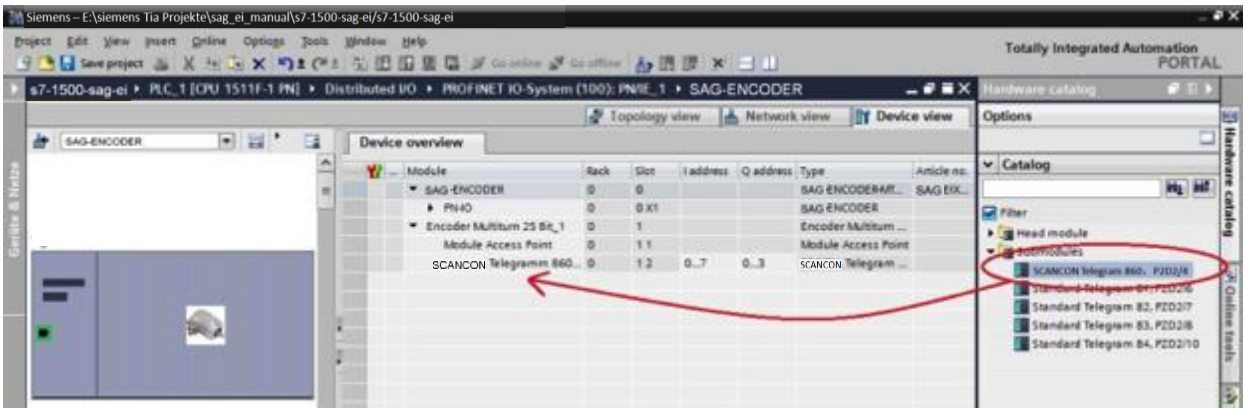
The transfer of the Base Mode Parameter Access request to the DO/DU parameter manager is done by writing the request data structure onto the Parameter Access Point (PAP) data record. When the write operation finishes, the parameter manager state machine is triggered according to the next Figure.

The transfer of the Base Mode Parameter Access response from the DO/DU parameter manager back to the client is done by reading the response data structure out of the Parameter Access Point (PAP) data record. The response to the read access is dependent on the internal state of the parameter manager according to the next Figure.

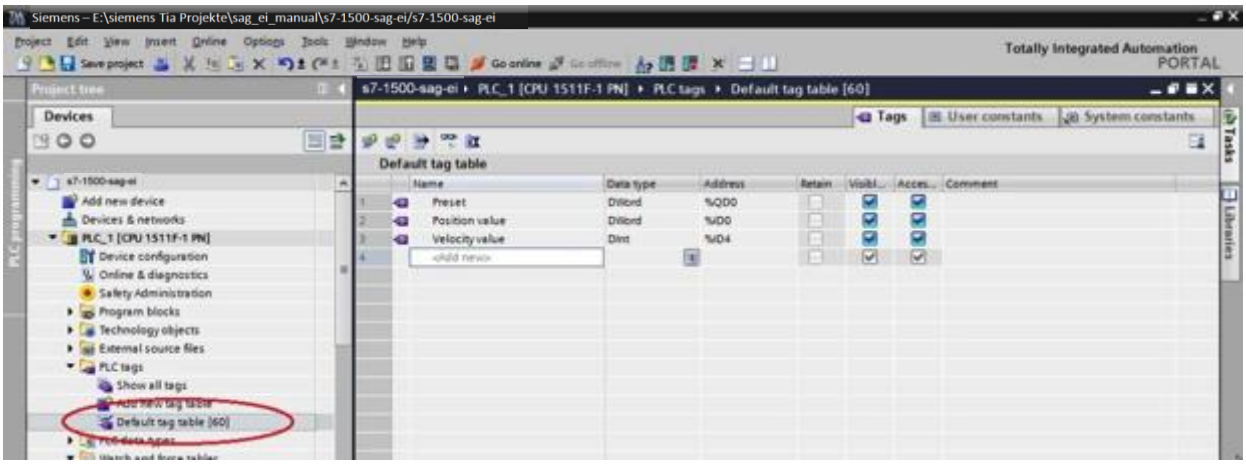


6. Configuring with TIA-Portal

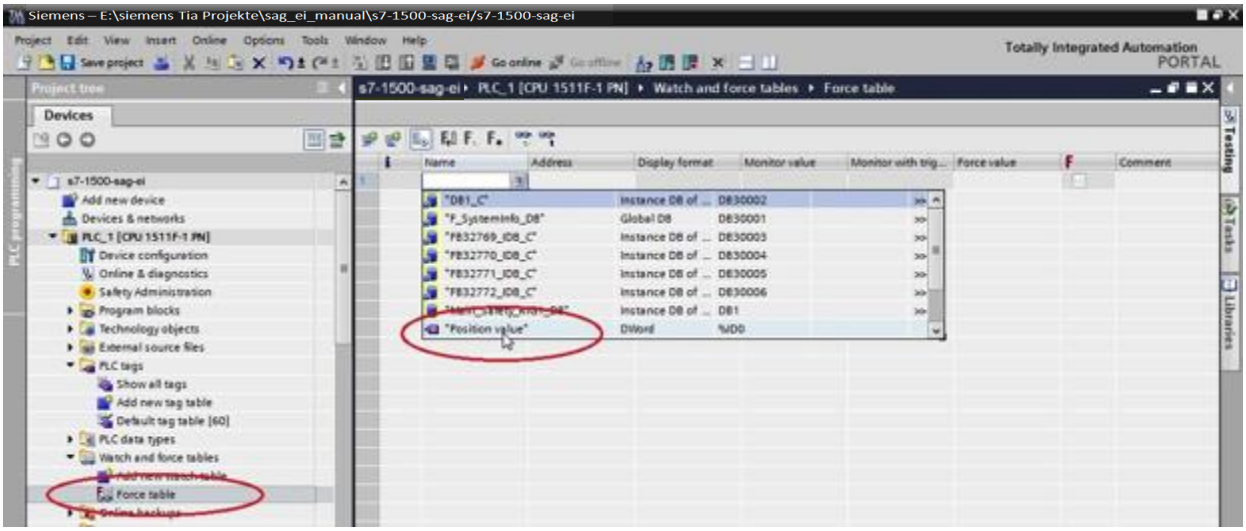
After installing the GSDML-file the related telegram can be set per drag and drop to the free slot.



Configure the default tags table

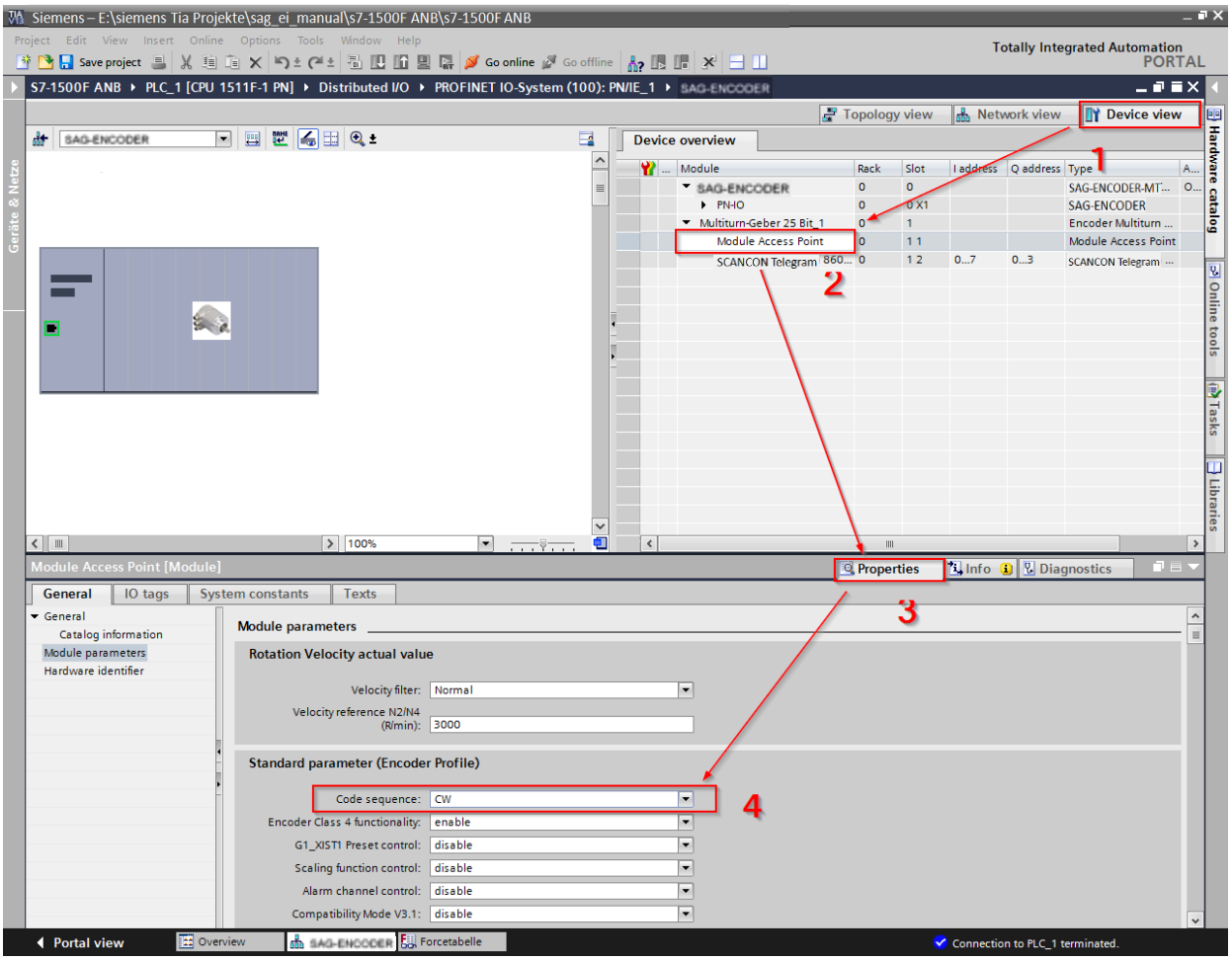


Force table



Force table to set i.e. the Preset value

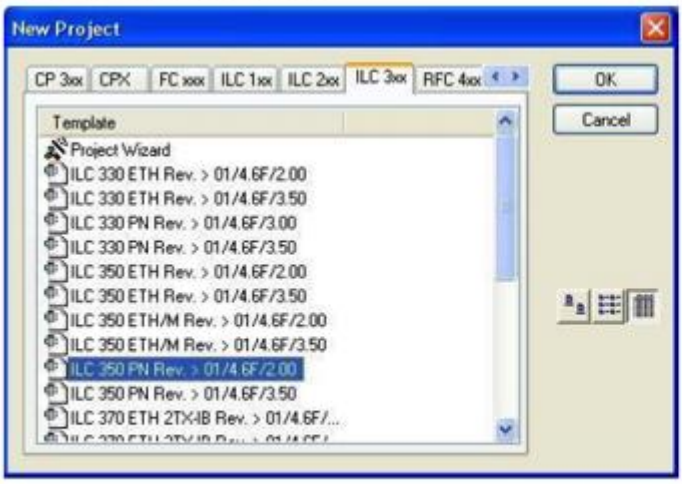
In the Module Access Point the parameters of the GSDML-file can be modified. This paramters will transmit to the encoder on each start of the PLC.



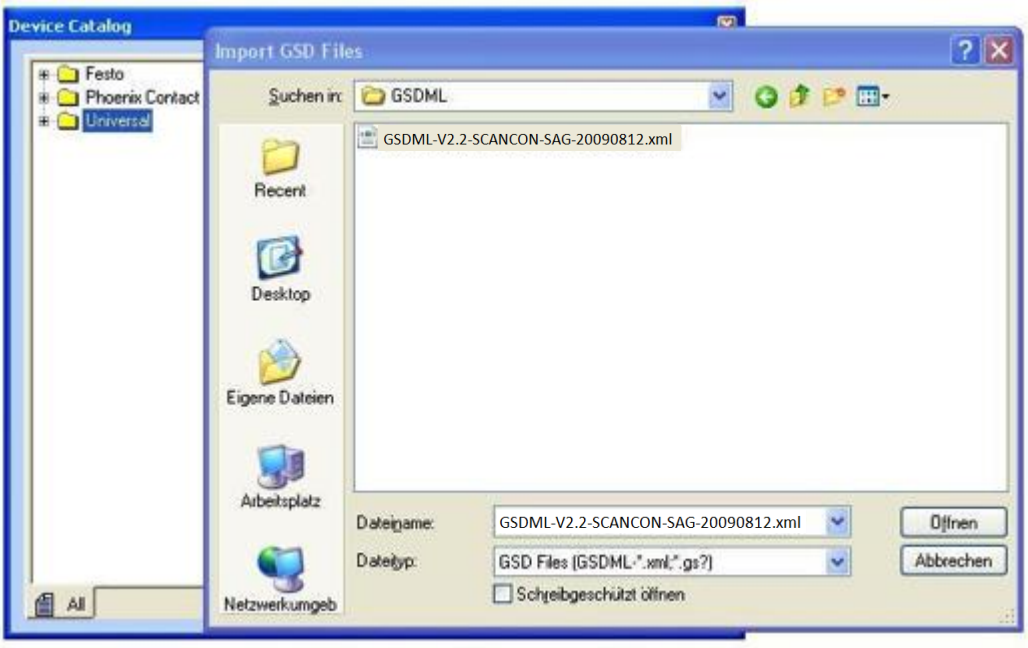
7. Configuring with PC Worx

In the following chapter the configuration of the SCANCON encoder with the configuration tool is shown exemplarily. In this example PC Worx Version 6.00.25 SP2.56 with workaround for GSDML import are used. If there are questions about details please contact the manufacturer.

Creating a New PNIO Project:



Installing the GSDML file



Insert the PROFINET IO SAG-Encoder below the PROFINET IO controller node.

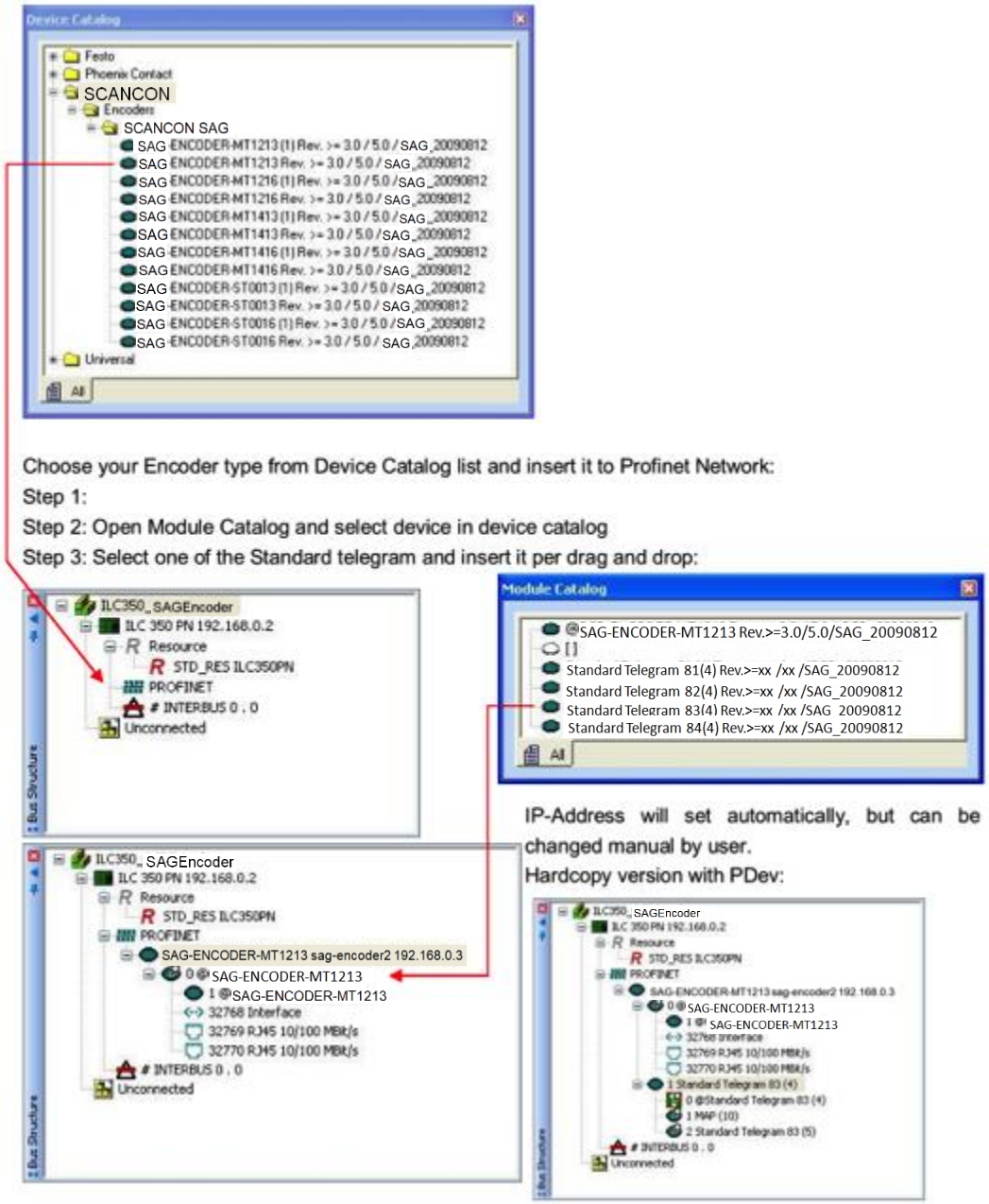
• If the device catalog is hidden, show it by selecting the "View/Device Catalog" menu.

• Open the "SCANCON" device catalog.

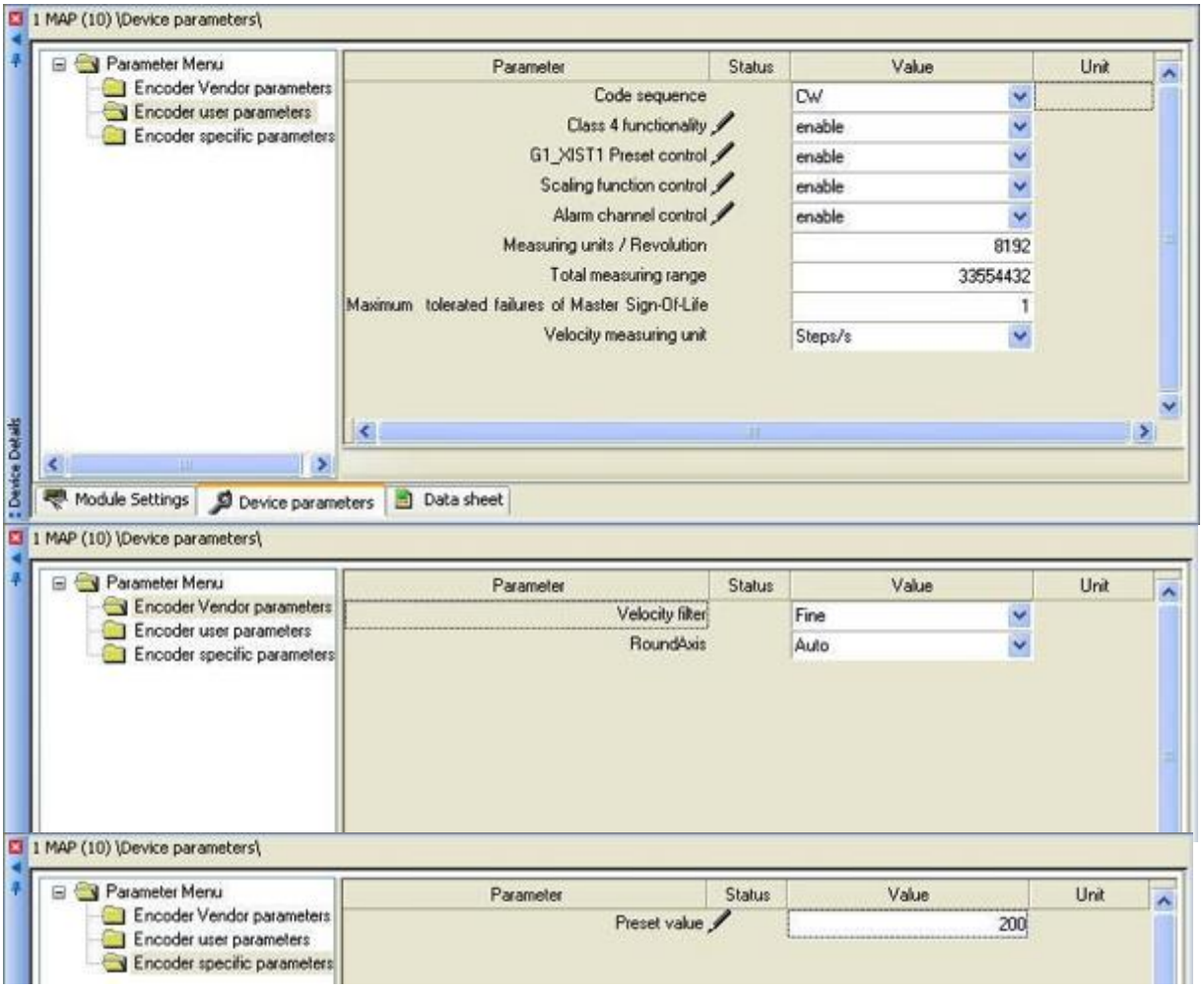
(MT = Multi-Turn, ST = Single-Turn, (1) without PDev = no IRT)

PDev necessary for extended setup (AutoCrossing, AutoNegoiation, FastStartUp, Topology for IRT (neighborhood detection, port setup))

Choose your Encoder type from Device Catalog list and insert it to Profinet Network:

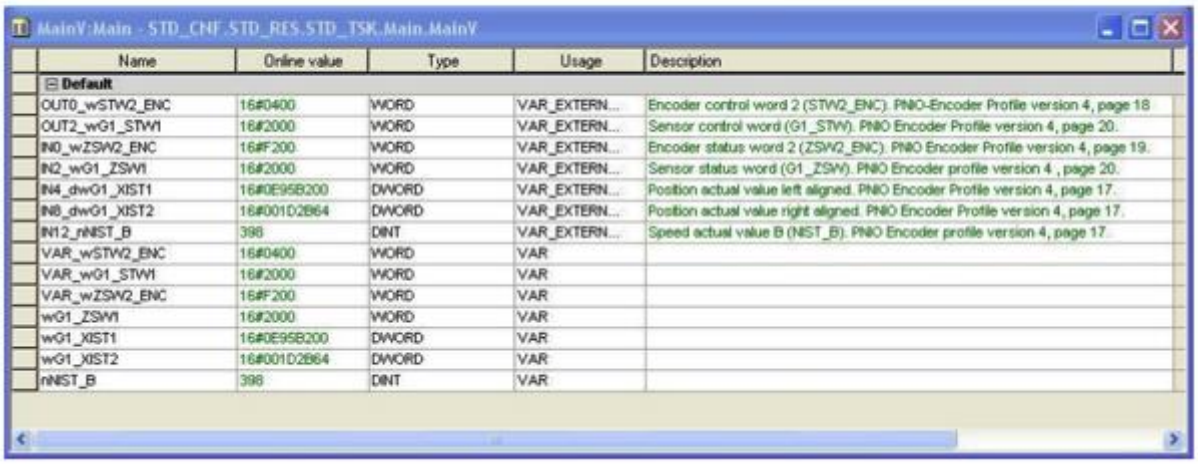


Setting Encoder Parameters in Device Details dialog:



Mapping Variable to the Standard telegram (I/O Data)

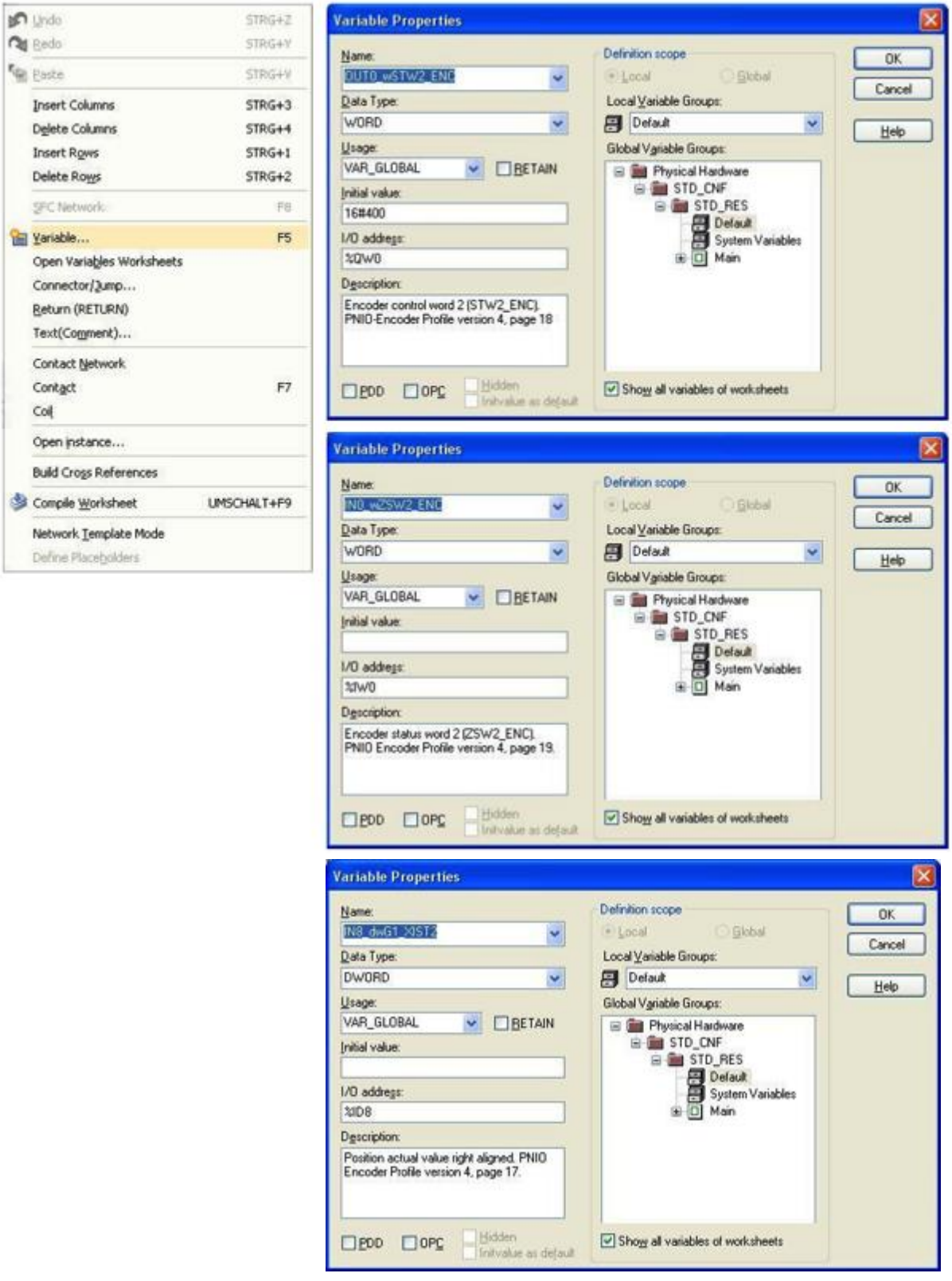
➔ Create new parameter table:



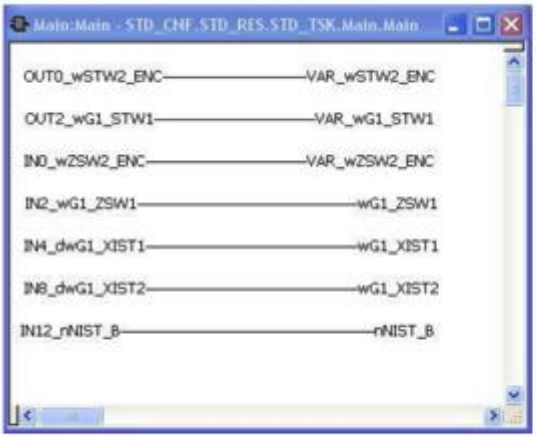
Name	Online value	Type	Usage	Description
Default				
OUT0_wSTW2_ENC	16#0400	WORD	VAR_EXTERN...	Encoder control word 2 (STW2_ENC). PNIO-Encoder Profile version 4, page 18
OUT2_wG1_STW1	16#2000	WORD	VAR_EXTERN...	Sensor control word (G1_STW). PNIO Encoder Profile version 4, page 20.
IN0_wZSW2_ENC	16#F200	WORD	VAR_EXTERN...	Encoder status word 2 (ZSW2_ENC). PNIO Encoder Profile version 4, page 19.
IN2_wG1_ZSW1	16#2000	WORD	VAR_EXTERN...	Sensor status word (G1_ZSW). PNIO Encoder profile version 4 , page 20.
IN4_dwG1_XIST1	16#0E95B200	DWORD	VAR_EXTERN...	Position actual value left aligned. PNIO Encoder Profile version 4, page 17.
IN8_dwG1_XIST2	16#001D2B64	DWORD	VAR_EXTERN...	Position actual value right aligned. PNIO Encoder Profile version 4, page 17.
IN12_nNIST_B	398	DINT	VAR_EXTERN...	Speed actual value B (NIST_B). PNIO Encoder profile version 4, page 17.
VAR_wSTW2_ENC	16#0400	WORD	VAR	
VAR_wG1_STW1	16#2000	WORD	VAR	
VAR_wZSW2_ENC	16#F200	WORD	VAR	
wG1_ZSW1	16#2000	WORD	VAR	
wG1_XIST1	16#0E95B200	DWORD	VAR	
wG1_XIST2	16#001D2B64	DWORD	VAR	
nNIST_B	398	DINT	VAR	

Sample:

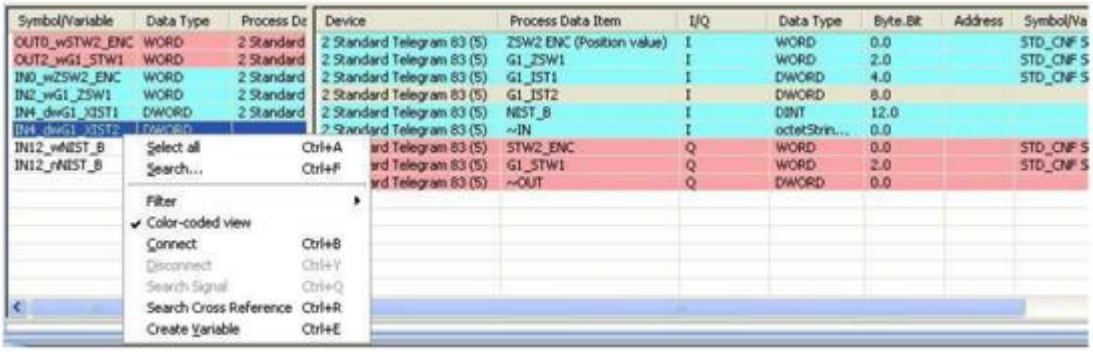
Right click and insert new Global variable and map to the I/O Address:



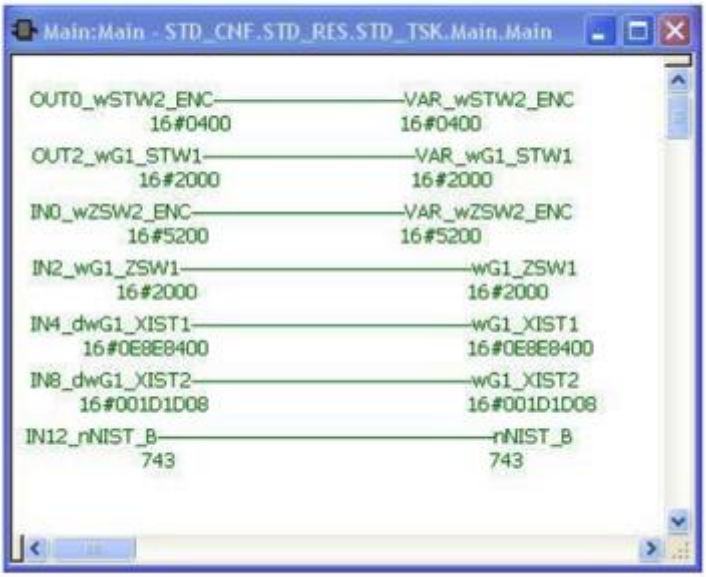
Create new Variable as Local and connect to the Mapped I/O Variable with drag and drop:



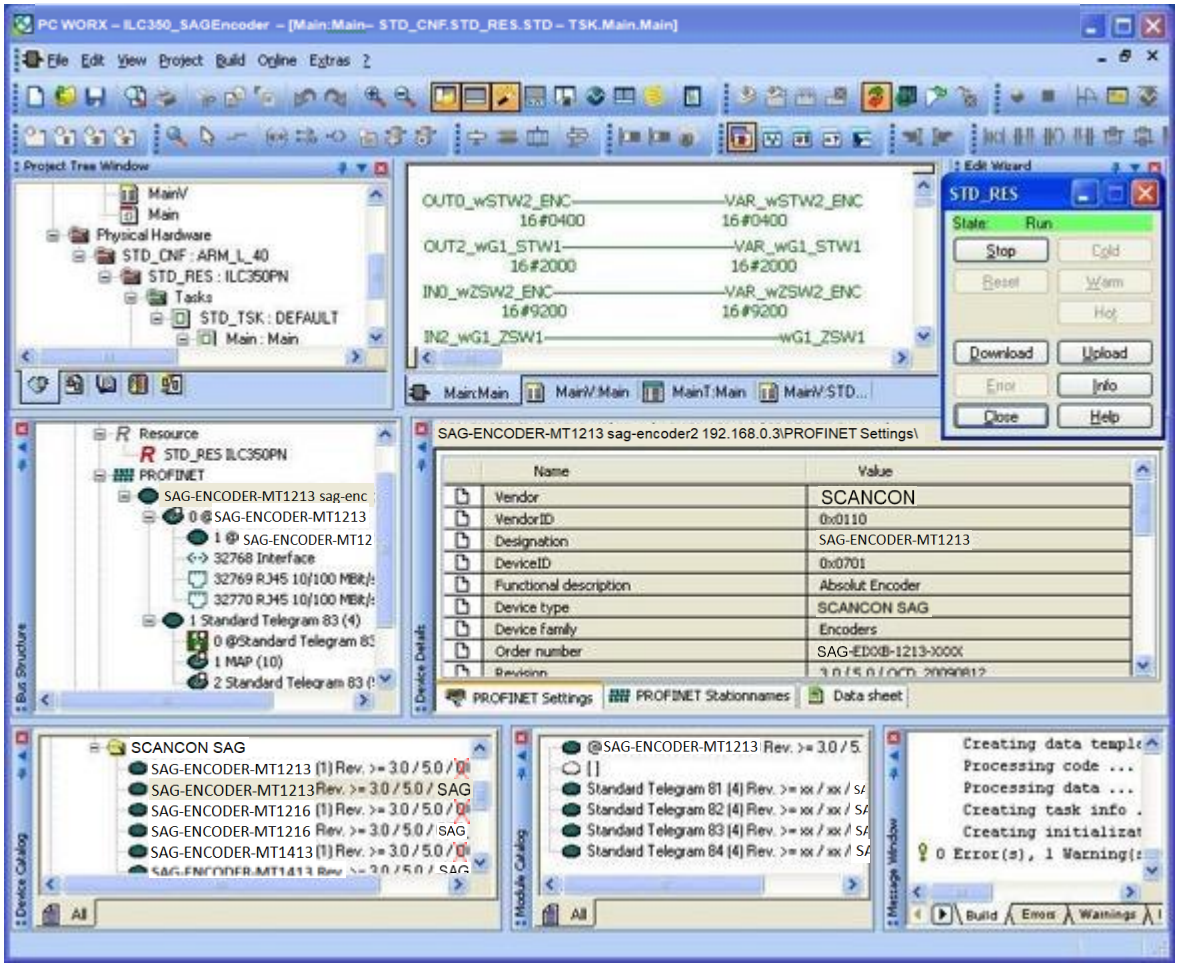
Assigning the Variables to the Encoder I/O in dialog Process Data assignment:



Mark the Variable and start to connect.
Sample: Online debugging mode



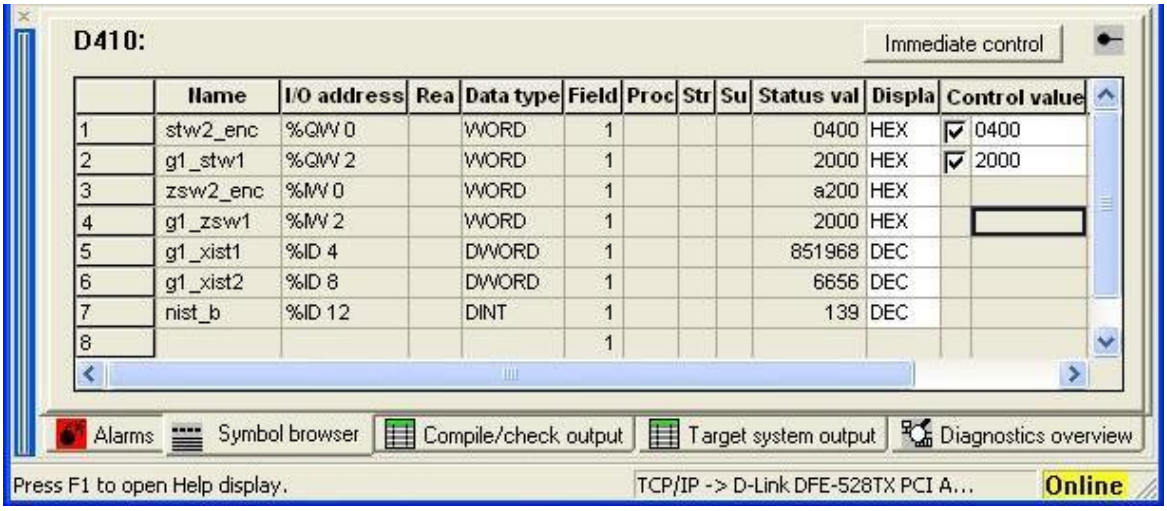
In the next hardcopy is available the complete running project:



NOTE: If some encoder parameter (i.e. Totalresolution) in the table 1 MAP device parameter missing, then contact PhoenixContact for an additional workaround.

8 FAQ

1. **Question:** Why don't I get back positions values in g1_xist2?
   Answer: According the encoder profile it is necessary to set Bit 10 to "1" in stw2 and bit 13 in g1_stw1. See the next hardcopy. Or an error is given and is not confirmed.



2. **Question:** Why don't work the neighboring detection?
   Answer: The encoder supports the LLDP protocol. But it is necessary to use the newest version of Step 7 or Simotion Scout. The flag "Device replacement without replacement medium" must be active in the Properties window under General.

3. **Question:** What is to do if one encoder has to replace by a new one?
   Answer: See answer 2 or chapter 4.3.

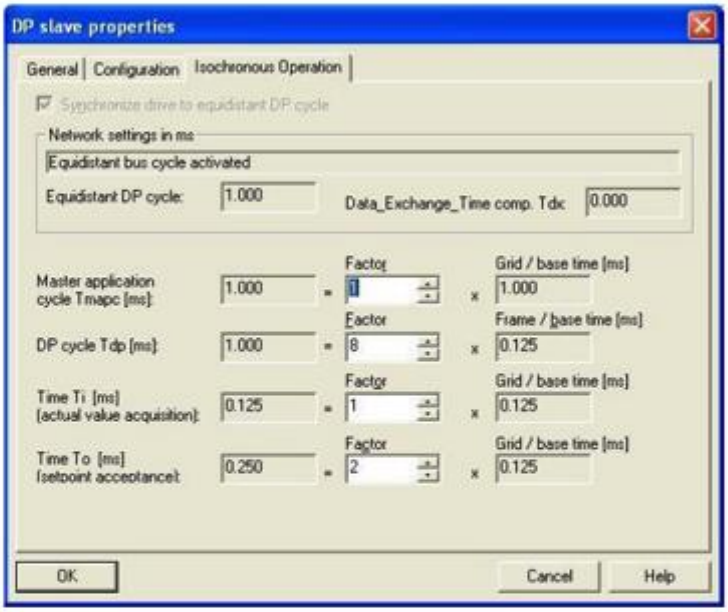
4. **Question:** In the application is a single-turn encoder in use. Can this replaced by a multi-turn encoder too and what is to do?
   Answer: There is nothing to do. A multi-turn can substitute a single-turn automatically.

5. **Question:** Why don't work the communication between encoder and PLC correct?
   Answer: The Firmware of the PLC and the STEP 7 (with minimum Hot fix 6) or Simotion Scout has to use the newest firmware that support IRT 2.2 or Stack version 3.1 for Ertec devices.

6. **Question:** What is the easiest way to set the preset value?
   Answer: Use Telegram 860. See chapter about Preset setting.

7. **Question:** Why can I not set the preset value or the other parameters?
   Answer: Only in class 3 with activated class 4 functionality or class 4 is it possible to set the parameters. If necessary it is important to use class 4 or to activate the class 4 functionality in the Hardware Manager.

8. **Question:** On using the D410 the error "Synchronization error between Profibus and Profinet" popped up. What is to do?
Answer: Both systems have to use the same cycle time. If the Profinet cycle time amounts 1ms then must use the Profibus the same time. See the next Hardcopy with the settings for 1ms.



9. **Question:** What is the different between Encoder Profil 4.0 and 4.1?
Answer:

	4.0	4.1
G_XIST1	Position value, left alligned	Counter value, right aligned
GSDML		
MAP Parameter	Inclusive Telegrams	Separate Telegrams

9 Technical data, accessories and type keys

These information are available on our data sheet. You can download it free of charge from the SCANCON website.

10 Glossar

Term	Explantations
10Base-T	Transmission line with 10 Mbit data transmission rate
100Base-T	Transmission line with 100 Mbit data transmission rate
Auto crossing	Allow to use straight or crossover wiring
Auto negotiation	Is an Ethernet procedure by which two connected devices choose common transmission parameters, such as speed and duplex mode
Baud rate	Transmission rate; it display the transmission bits per second
Binary	Numeric system with value 0 or 1.
CAT5	Terminations for transmission rates up to 100 Mbit.
DCP.Hello	On Fast Start up the encoder will register to the IO-Controller with the "DCP.Hello"-Service
EMC	**Electromagnetic compatibility**, there are rules to verifying devices.
Ethernet	Ethernet is a computer network technology based on frames.
Endless shaft	(Round axis) Solve the problem with not binary values for revolutions
Fast Ethernet	Transmission technology with 100 Mbit transmission rate.
Fast Start up	Optimized start up time for Profinet (< 1s)
Flash	Internal memory, saved data will be available after power down.
GSDML	**G**eneric **S**tation **D**escription **M**arkup **L**anguage: XML based description language. Contains all available parameters, classes, …
Implicit Messaging	IO Connection: communication between controller and device
IP-Address	Allow a logic addressing from computer in a network.
IRT flex	Former name for the IRT synchronization "High Flexibility"
IRT top	Former name for the IRT synchronization "High Performance"
LLDP	**L**ink **L**ayer **D**iscovery **P**rotocol
MAC Address	Worldwide explicit address of a device. The encoder uses three MAC Addresses: one for internal interface and two for the ports. The basic MAC Address is available on the type label.
Mbit	Transmission rate or baud rate, million bits per second
MAP	**M**odule **A**ccess **P**oint. This MAP Sub module contains at least the mandatory Parameter Access Point (PAP) which is mapped to a dedicated Record Data Object
SAG	Acronym: **OPTOCODE**, name of an encoder series manufactured by SCANCON DK.
OSI-Model	The **O**pen **S**ystem **I**nterconnection reference model is a open layer model for the organization of a communication.
PDEV	**P**hysical **dev**ice. Not all PLC's support several sub slots. Then select in the product tree Customized otherwise ProfileV4.x
Round Axis	See -> Endless shaft

Term	Explanations
Switch	A switch is an electronic device to connect computers e.g. network segments in a local network. Unlike a hub, a switch uses stacks to avoid network collisions.
TCP	The **T**ransmission **C**ontrol **P**rotocol is a connection orientated transmission protocol, in a network.
UDP	**U**ser **D**atagram **P**rotocol is utilized to send data that does not need to be transferred in a reliable way.

11 Revision index

Revision	Date	Revision
First release	5.6.2008	2.00
Add mechanical drawings, change some details	6.3.2009	2.01
Several small corrections	8.3.2009	2.02
G1_XIST3 -> G1_XIST1 for Telegram 81-83	27.3.2009	2.03
Add FAQ and detailed info about IRT configuration	29.4.2009	2.04
Update the information about “Device replacement without replacement medium”	15.5.2009	2.05
Update information about Preset setting, update Encoder functions, CD, PDEV	27.5.2009	2.06
Update Preset details	1.7.2009	2.07
Fast startup, DCP Hello, Configuration on PCWorx	19.11.2009	2.08
Delete information about a pause during round axis functionality, update of the IRT functionality	27.10.2010	2.09
Updated type keys on front page, removed content of sections “technical data” and “accessories and documentation”	29.02.2016	2.10
Add short instruction about TIA-Portal	2.1.2017	2.11