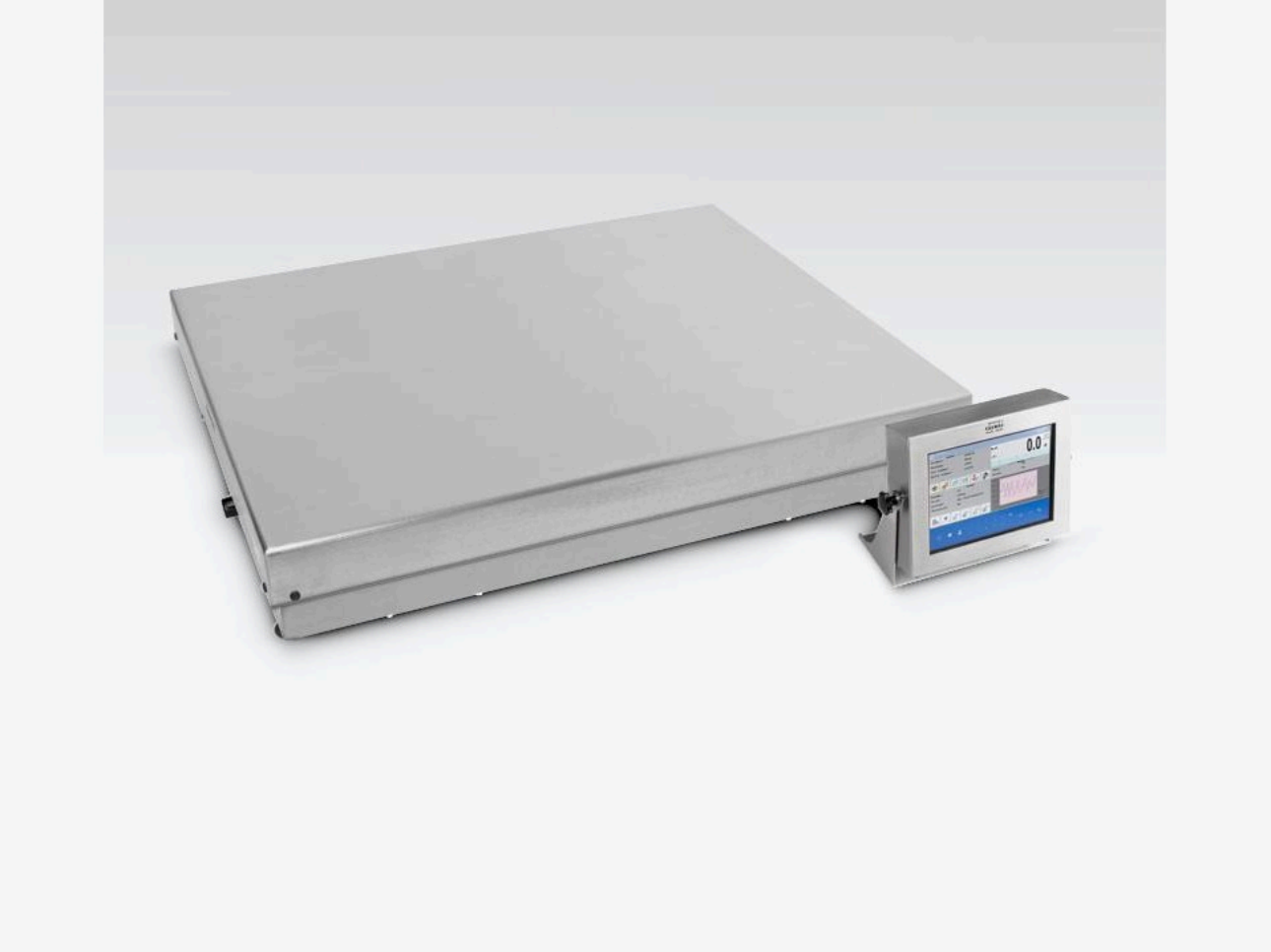
USER GUIDE

HY10.300.1.HRP.H

Radwag

Distributed by AKRIBIS



Software manual

ITKP-05-02-07-21-EN

PROFINET

Communication Protocol of PUE HY10 Terminal

radwag.com
Scan the QR code to view additional scientific material that may be of interest to you.
There you will find more useful information in an accessible format!

AKRIBIS www.akribis.info contacto@akribis.info

JULY 2021

CONTENTS

1. DATA STRUCTURE 4

1.1. Input Address 4

1.2. Output Address 7

2. CONFIGURATION OF PROFINET MODULE IN TIA PORTAL V14 9

2.1. GSD Import 10

2.2. Module Configuration 11

3. PLC SOFTWARE SAMPLE 15

1. DATA STRUCTURE

1.1. Input Address

Input variables list:

Variable	Offset	Length [WORD]	Data type
Platform 1 mass	0	2	float
Platform 1 tare	4	2	float
Platform 1 unit	8	1	word
Platform 1 status	10	1	word
Platform 1 LO threshold	12	2	float
Platform 2 mass	16	2	float
Platform 2 tare	20	2	float
Platform 2 unit	24	1	word
Platform 2 status	26	1	word
Platform 2 LO threshold	28	2	float
Platform 3 mass	32	2	float
Platform 3 tare	36	2	float
Platform 3 unit	40	1	word
Platform 3 status	42	1	word
Platform 3 LO threshold	44	2	float
Platform 4 mass	48	2	float
Platform 4 tare	52	2	float
Platform 4 unit	56	1	word
Platform 4 status	58	1	word
Platform 4 LO threshold	60	2	float
Process status (Stop, Start)	64	1	word
Inputs status	66	1	word
Min	68	2	float
Max	72	2	float
Min2	76	2	float
Max2	80	2	float
Lot number	84	2	dword
Operator	88	1	Word
Product	90	1	word
Customer	92	1	word

Packaging	94	1	word
Source warehouse	96	1	word
Target warehouse	98	1	word
Formulation/Dosing	100	1	word

Platform mass – response: platform mass in current unit.

Platform tare – response: platform tare in adjustment unit.

Platform unit – determines currently displayed mass unit of a platform.

Unit bits	
0	Gram [g]
1	Kilogram [kg]
2	Carat [ct]
3	Pound [lb]
4	Ounce [oz]
5	Newton [N]

Example:

bit No.	**B5**	**B4**	**B3**	**B2**	**B1**	**B0**
value	0	0	0	0	1	0

The unit of the weighing instrument is kilogram [kg].

Platform status – determines status of a weighing platform.

Status bits	
0	measurement correct (weighing instrument does not report an error)
1	stable measurement
2	weighing instrument indicates zero
3	weighing instrument is tared
4	weighing instrument is in II weighing range
5	weighing instrument is in III weighing range
6	weighing instrument reports NULL error
7	weighing instrument reports LH error
8	weighing instrument reports FULL error

Example:

bit No.	**B8**	**B7**	**B6**	**B5**	**B4**	**B3**	**B2**	**B1**	**B0**
value	0	0	0	0	1	0	0	1	1

The weighing instrument does not report error, stable measurement in II weighing range.

LO threshold – response: **LO** threshold value of a platform in adjustment unit.

Process status – determines process status:

Decimal value	**Process status**	**bit No.**	
		B1	**B0**
0	process disabled	0	0
1	Process start	0	1
2	process stop	1	0
3	process completed	1	1

Inputs status – response: status of set inputs:

Input No.	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**
OFF	0	0	0	0	0	0	0	0	0	0	0	0
ON	1	1	1	1	1	1	1	1	1	1	1	1

Example:

Mask of set 2 and 4 inputs: 0000 0000 0000 1010

MIN – response: **MIN** threshold value (in the current unit selected for active working mode).

MAX – response: **MAX** threshold value (in the current unit selected for active working mode).

Lot number – response: lot number.

Operator – response: code of logged in operator.

Product – response: code of selected product.

Customer – response: code of selected customer.

Packaging – response: code of selected packaging.

1.2. Output Address

Input variables list:

Variable	Offset	Length [WORD]	Data type
Command	0	1	word
Command with parameter	2	1	word
Platform	4	1	word
Tare	6	2	float
Lo threshold	10	2	float
Outputs status	14	1	word
Min	16	2	float
Max	20	2	float
Min2	24	2	float
Max2	28	2	float
Lot number	32	2	dword
Operator	36	1	word
Product	38	1	word
Customer	40	1	word
Packaging	42	1	word
Source warehouse	-	-	-
Target warehouse	-	-	-
Formulation/Dosing	48	1	word

Basic command – setting respective value performs the task in accordance with the table:

Decimal value	Command
1	Zero the platform
2	Tare the platform
8	Delete statistics
16	Save/Print
32	Start
64	Stop (error)

Example:

16# 0020 – command carried out, response: process start.

Complex command – setting respective value performs the task in accordance with the table:

Decimal value	Command
1	Setting the tare value for a given platform
2	Setting LO threshold value for a given platform
4	Setting outputs status
8	Setting MIN threshold value
16	Setting MAX threshold value
32	Setting user value 1
64	Zero given platform
128	Tare given platform
256	Setting active platform
512	Setting MIN2 threshold value
1024	Setting MAX2 threshold value

Complex command requires setting address of respective parameter (from 2 to 24 – refer to: 'Complex command parameters' table).

Example:

16#0002 – command sets LO threshold to the value set in LO parameter (address 5 – refer to: *'Complex command parameters'* table).

Platform – complex command parameter: weighing platform number.

Tare – complex command parameter: tare value (in adjustment unit).

LO threshold – complex command parameter: LO threshold value (in adjustment unit).

Outputs status – complex command parameter: determines status of weighing indicator outputs.

Output No.	12	11	10	9	8	7	6	5	4	3	2	1
OFF	0	0	0	0	0	0	0	0	0	0	0	0
ON	1	1	1	1	1	1	1	1	1	1	1	1

Example:

Mask of active 2 and 4 outputs: 0000 0000 0000 1010

MIN – complex command parameter: MIN threshold value (in the current unit selected for active working mode).

MAX – complex command parameter: MAX threshold value (in the current unit selected for active working mode).

Lot number – complex command parameter: lot number.

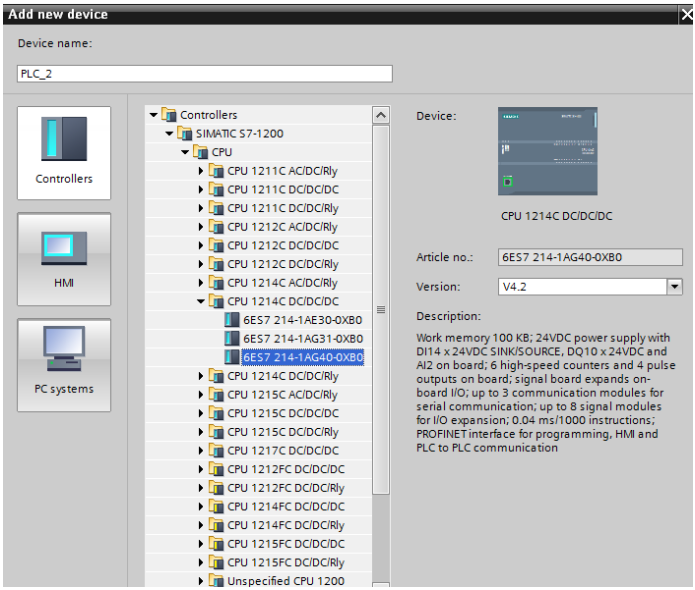
⚠	***A command or a command with parameter is executed once when its bit setting is detected. If the command with the same bit is to be executed again, zero the bit.***

Example:

Command	
Taring	0000 0000 0000 0010
Command bits zeroing	0000 0000 0000 0000
Taring	0000 0000 0000 0010

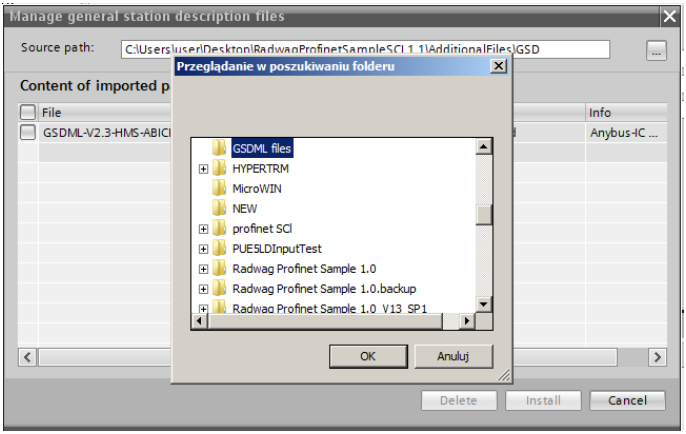
2. CONFIGURATION OF PROFINET MODULE IN TIA PORTAL V14

Operating the environment has to be preceded with creating a new project in which the topology of the PROFINET network with MASTER device is determined (in this example: SIEMENS S7-1200).

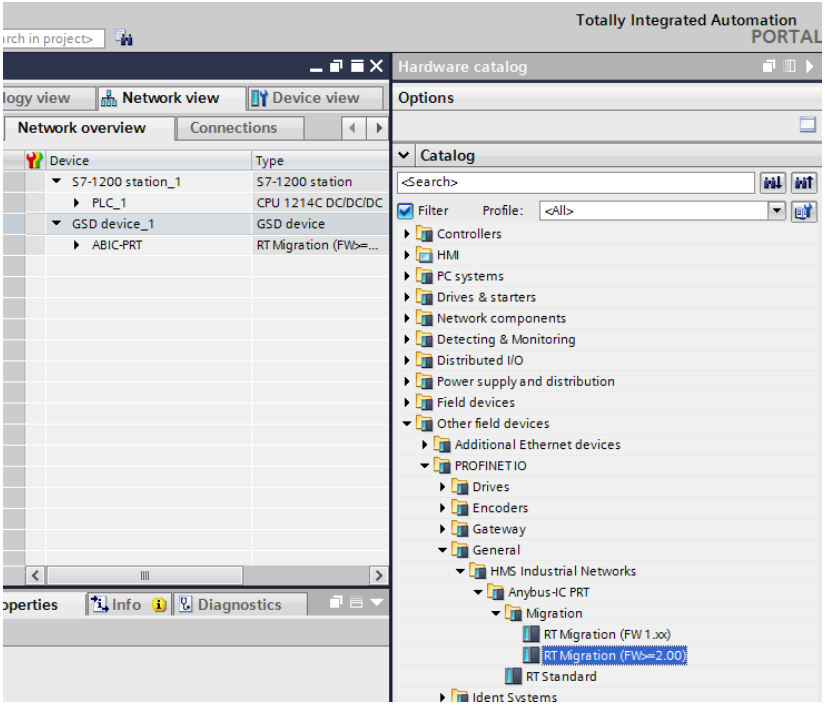


2.1. GSD Import

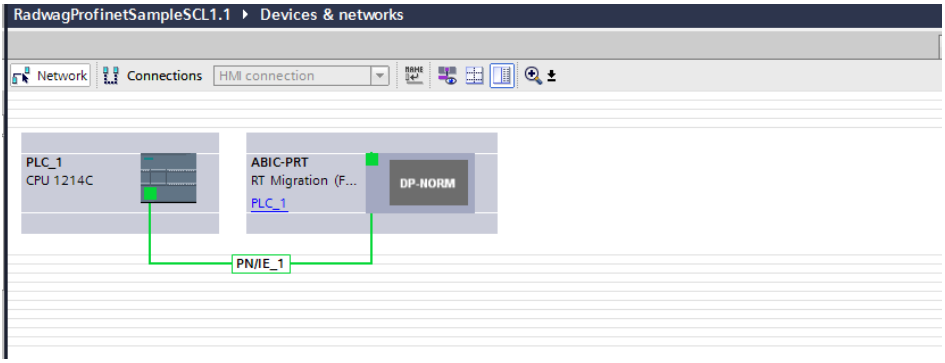
Using the included GSD configuration file add new device to the environment. Use OPTIONS tab first, MANAGE GENERAL STATION DESCRIPTION FILES (GSD) next and indicate the path to GSD file.



Upon successful adding of the file using list of devices, find ABIC-PRT module:

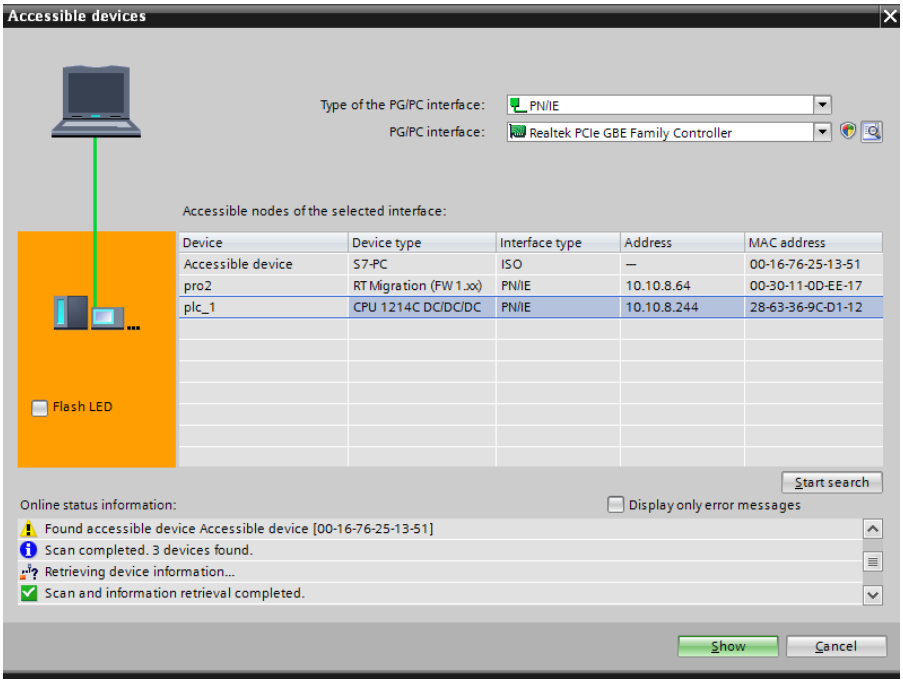


You can now create a network consisting of one MASTER PLC and added SLAVE module:

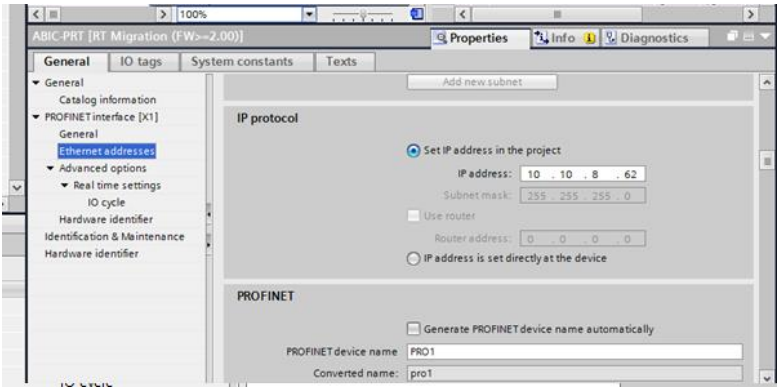


2.2. Module Configuration

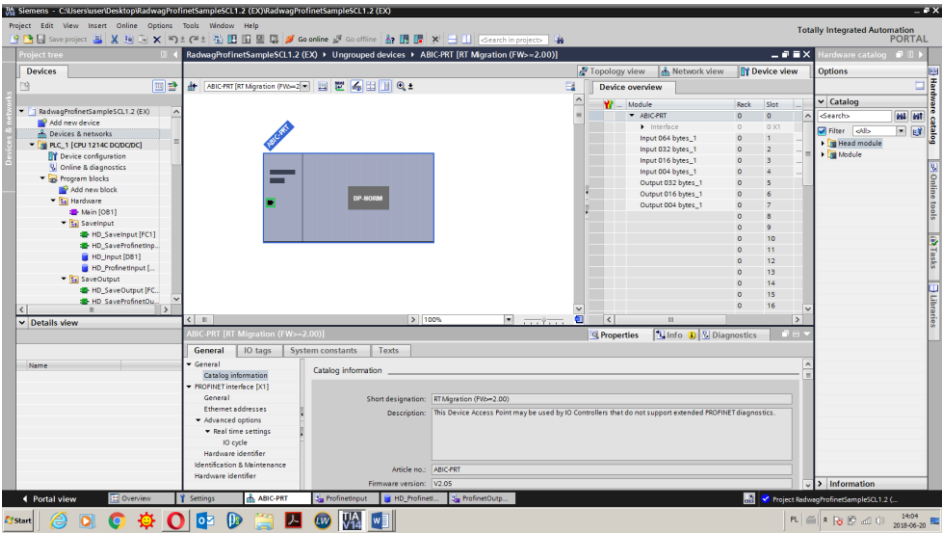
At this stage, create a network consisting of MASTER device and SLAVE device (weighing instrument). Upon connecting the power supply, search for device using ACCESSIBLE DEVICES function. The list should contain MASTER and SLAVE devices:

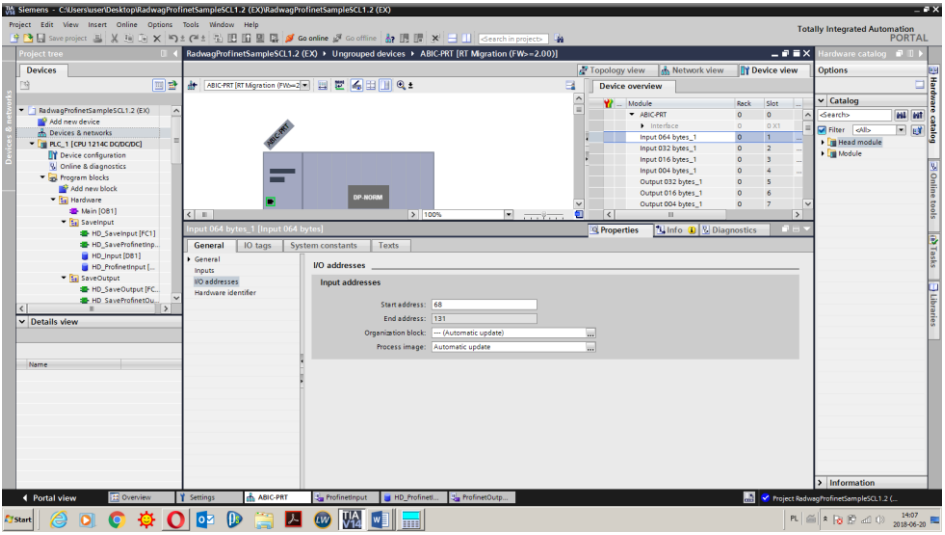


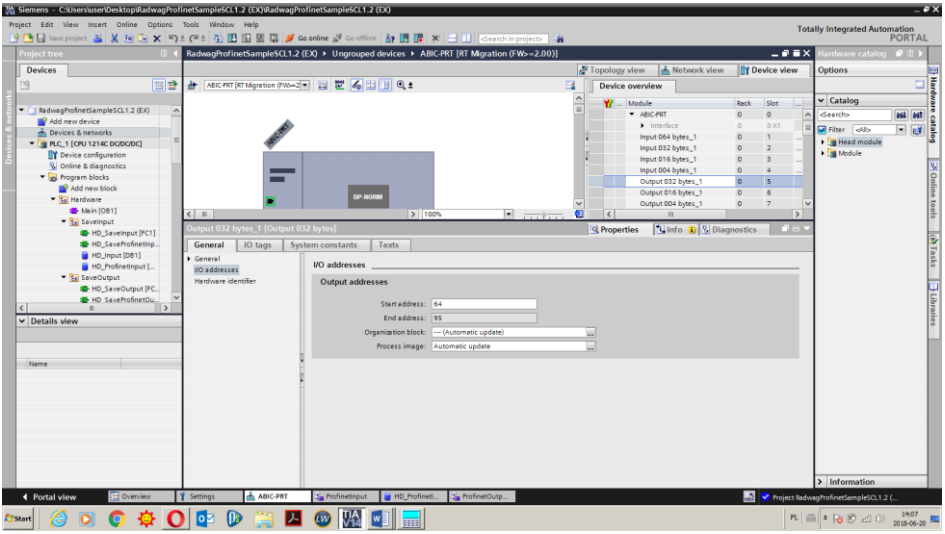
Next, specify the IP address of the module and its name in PROFINET network. Upon selecting the module in PROPERTIES tab, find PROFINET INTERFACE and enter IP address and name. Those settings have to be the same as the ones set in the weighing instrument menu. IP SLAVE address has to be part of the same subnet as the MASTER address.



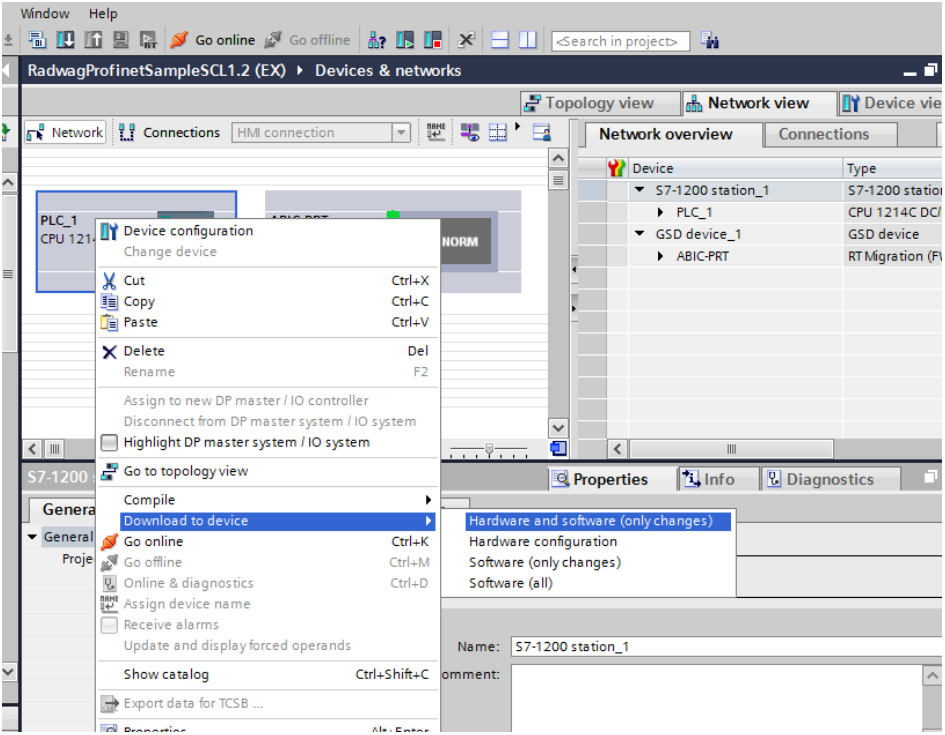
Proceed to module configuration. Start by determining the size and the starting address of input and output registers. To do this, select modules from the list of INPUT and OUTPUT modules as in the picture below. The maximum size of the input and output data is 116 bytes each. Default starting addresses were used in the project - 68 for INPUT module and 64 for OUTPUT module:



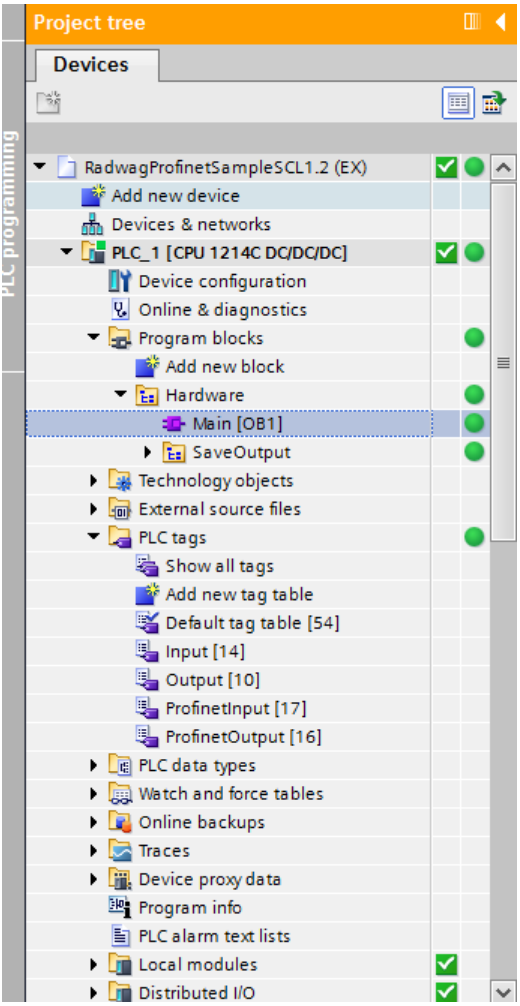

Siemens - C:\Users\user\Desktop\RadwagProfinetSampleSCL1.2 (EX)\RadwagProfinetSampleSCL1.2 (EX)
Totally Integrated Automation PORTAL
RadwagProfinetSampleSCL1.2 (EX) ▸ Ungrouped devices ▸ ABIC-PRT [RT Migration (FW>=2.00)]
Input 064 bytes_1 [Input 064 bytes]
I/O addresses
Input addresses
Start address: 68
End address: 131
Organization block: --- (Automatic update)
Process image: Automatic update


Siemens - C:\Users\user\Desktop\RadwagProfinetSampleSCL1.2 (EX)\RadwagProfinetSampleSCL1.2 (EX)
Totally Integrated Automation PORTAL
RadwagProfinetSampleSCL1.2 (EX) ▸ Ungrouped devices ▸ ABIC-PRT [RT Migration (FW>=2.00)]
Output 032 bytes_1 [Output 032 bytes]
I/O addresses
Output addresses
Start address: 64
End address: 95
Organization block: --- (Automatic update)
Process image: Automatic update

On this stage you can download hardware and software configuration to the device and download data to the device.



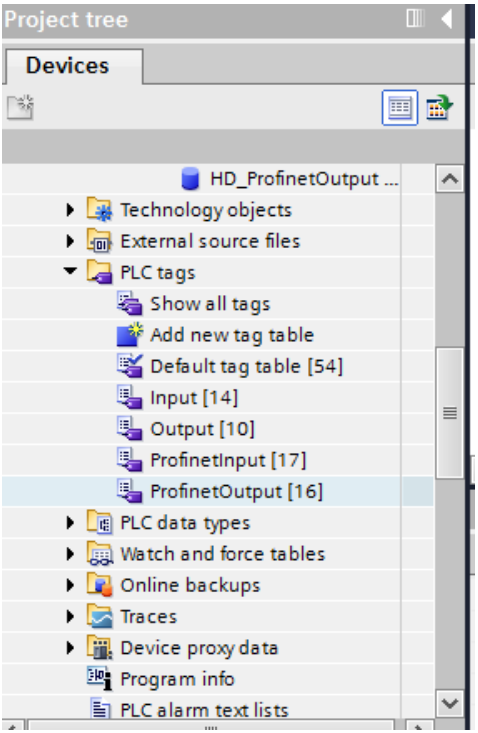
Upon successful compilation and loading of the code, MASTER and SLAVE modules should establish communication. You can check this by pressing GO ONLINE field. The result has to be similar to the result presented below.



The next step will be to create program code.

3. PLC SOFTWARE SAMPLE

Start creating the application by determining symbolic names of input and output registers. For this purpose, use the branch of the project tree: PLC TAGS. For the purpose of this example, the figure below contains exemplary tags tables:



INPUT and OUTPUT tables refer to the physical inputs/outputs of the MASTER device and are not relevant in terms of this application. The PROFINET module input and output registers are specified in ProfinetInput and ProfinetOutput tables. The pictures below present determined symbolic names and addresses:

RadwagProfinetHY10 1.0 ang ▸ PLC_1 [CPU 1214C DC/DC/DC] ▸ PLC tags ▸ ProfinetInput [32]

Tags | User constants

ProfinetInput

	Name	Data type	Address	Retain	Acces...	Writa...	Visibl...	Comment
1	mass	Real	%ID68		☑	☑	☑	
2	tare	Real	%ID72		☑	☑	☑	
3	unit	Word	%IW76		☑	☑	☑	
4	status	Word	%IW78		☑	☑	☑	
5	LO	Real	%ID80		☑	☑	☑	
6	mass_2	Real	%ID84		☑	☑	☑	
7	tare_2	Real	%ID88		☑	☑	☑	
8	unit_2	Word	%IW92		☑	☑	☑	
9	status_2	Word	%IW94		☑	☑	☑	
10	LO_2	Real	%ID96		☑	☑	☑	
11	mass_3	Real	%ID100		☑	☑	☑	
12	tare_3	Real	%ID104		☑	☑	☑	
13	unit_3	Word	%IW108		☑	☑	☑	
14	status_3	Word	%IW110		☑	☑	☑	
15	LO_3	Real	%ID112		☑	☑	☑	
16	mass_4	Real	%ID116		☑	☑	☑	
17	tare_4	Real	%ID120		☑	☑	☑	
18	unit_4	Word	%IW124		☑	☑	☑	
19	status_4	Word	%IW126		☑	☑	☑	
20	LO_4	Real	%ID128		☑	☑	☑	

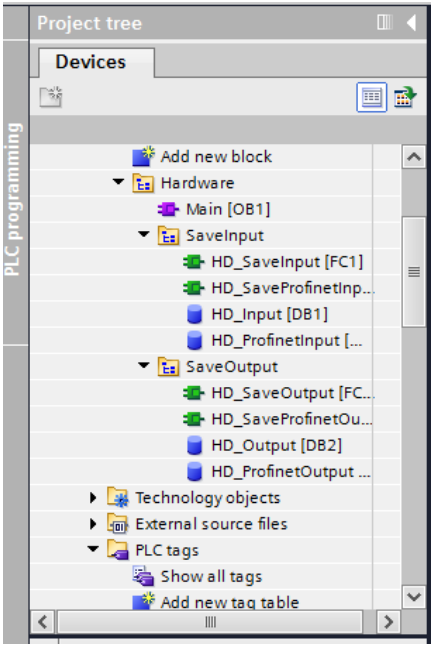
RadwagProfinetHY10 1.0 ang ▸ PLC_1 [CPU 1214C DC/DC/DC] ▸ PLC tags ▸ ProfinetInput [32]

Tags | User constants

ProfinetInput

	Name	Data type	Address	Retain	Acces...	Writa...	Visibl...	Comment
21	process_status	Word	%IW132		✓	✓	✓	
22	inputs	Word	%IW134		✓	✓	✓	
23	min	Real	%ID136		✓	✓	✓	
24	max	Real	%ID140		✓	✓	✓	
25	lot_number	DWord	%ID152		✓	✓	✓	
26	operator	Word	%IW156		✓	✓	✓	
27	article	Word	%IW158		✓	✓	✓	
28	customer	Word	%IW160		✓	✓	✓	
29	packaging	Word	%IW162		✓	✓	✓	
30	source_warehuse	Word	%IW164		✓	✓	✓	
31	target_warehouse	Word	%IW166		✓	✓	✓	
32	formulation/dosing_process	Word	%IW168		✓	✓	✓	

In order to avoid working directly on module physical inputs/outputs, create data blocks containing the representations of those registers and create function 'rewriting' the values between them. Create HARDWARE group in PROGRAM BLOCKS branch and determine data blocks in the same way as presented below:



HD_OUTPUT and HD_INPUT blocks refer to physical MASTER inputs/outputs and are not relevant in terms of this project. HD_ProfinetOutput and HD_ProfinetInput blocks refer to the PROFINET module input/output registers on a weighing instrument. They look as follows:

RadwagProfinetHY10 1.0 ang ▸ PLC_1 [CPU 1214C DC/DC/DC] ▸ Program blocks ▸ Hardware ▸ SaveInput ▸ HD_ProfinetInput [DB3]

Keep actual values | Snapshot | Copy snapshots to start values | Load start values as actual values

HD_ProfinetInput

	Name	Data type	Start value	Monitor value	Retain	Accessible f...	Writa...	Visible in ...	Setpoint	Com...
1	Static									
2	mass	Real	0.0	0.0						
3	tare	Real	0.0	0.0						
4	unit	Word	16#0	16#0000						
5	status	Word	16#0	16#0000						
6	lo	Real	0.0	0.0						
7	mass_2	Real	0.0	0.0						
8	tare_2	Real	0.0	0.0						
9	unit_2	Word	16#0	16#0000						
10	status_2	Word	16#0	16#0000						
11	lo_2	Real	0.0	0.0						
12	mass_3	Real	0.0	0.0						
13	tare_3	Real	0.0	0.0						
14	unit_3	Word	16#0	16#0000						
15	status_3	Word	16#0	16#0000						
16	lo_3	Real	0.0	0.0						
17	mass_4	Real	0.0	0.0						
18	tare_4	Real	0.0	0.0						
19	unit_4	Word	16#0	16#0000						
20	status_4	Word	16#0	16#0000						
21	lo_4	Real	0.0	0.0						

RadwagProfinetHY10 1.0 ang ▸ PLC_1 [CPU 1214C DC/DC/DC] ▸ Program blocks ▸ Hardware ▸ SaveInput ▸ HD_ProfinetInput [DB3]

Keep actual values | Snapshot | Copy snapshots to start values | Load start values as actual values

HD_ProfinetInput

	Name	Data type	Start value	Monitor value	Retain	Accessible f...	Writa...	Visible in ...	Setpoint	Com...
22	process_status	Word	16#0	16#0000						
23	inputs	Word	16#0	16#0000						
24	min	Real	0.0	0.0						
25	max	Real	0.0	0.0						
26	lot_number	DWord	16#0	16#FFFF_FFFF						
27	operator	Word	16#0	16#FFFF						
28	article	Word	16#0	16#0001						
29	customer	Word	16#0	16#FFFF						
30	packaging	Word	16#0	16#FFFF						
31	source_warehouse	Word	16#0	16#FFFF						
32	target_warehouse	Word	16#0	16#FFFF						
33	formulation/dosing_process	Word	16#0	16#FFFF						

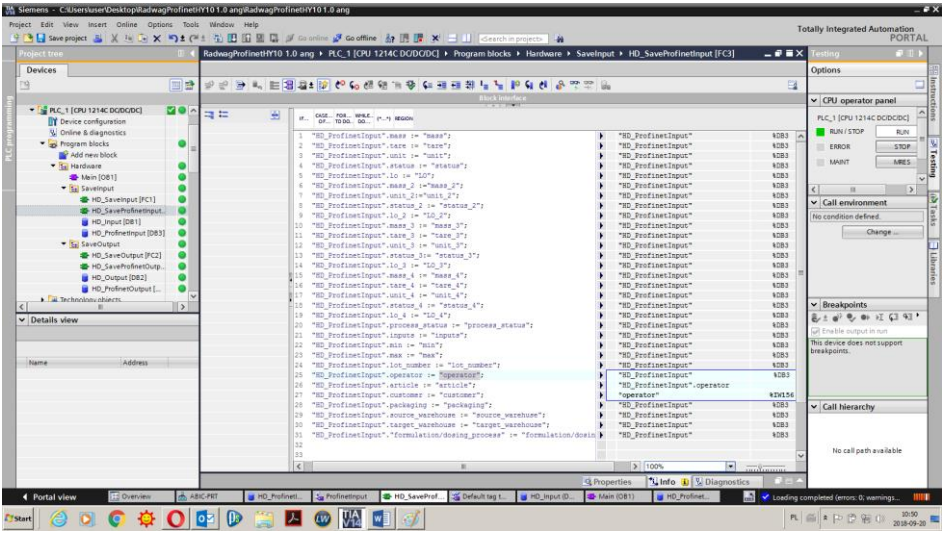
RadwagProfinetSampleSCL1.2 (EX) ▸ PLC_1 [CPU 1214C DC/DC/DC] ▸ Program blocks ▸ Hardware ▸ SaveOutput ▸ HD_ProfinetOutput [DB4]

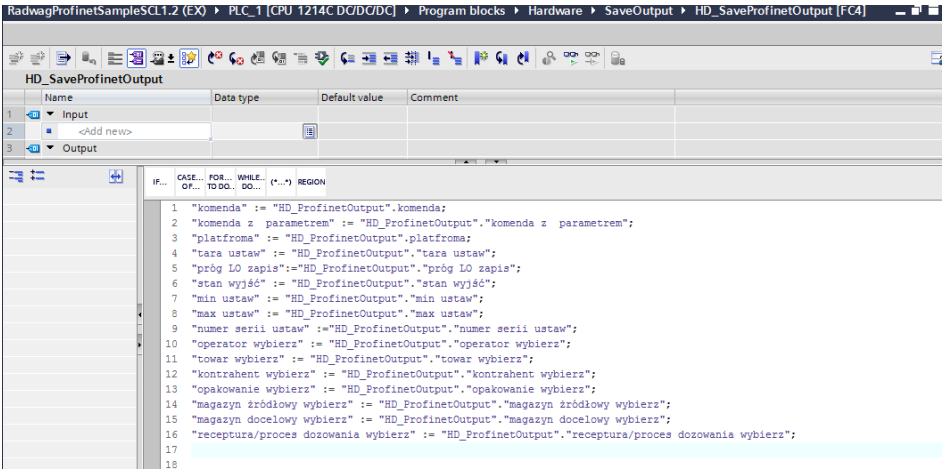
Keep actual values | Snapshot | Copy snapshots to start values | Load start values as actual values

HD_ProfinetOutput

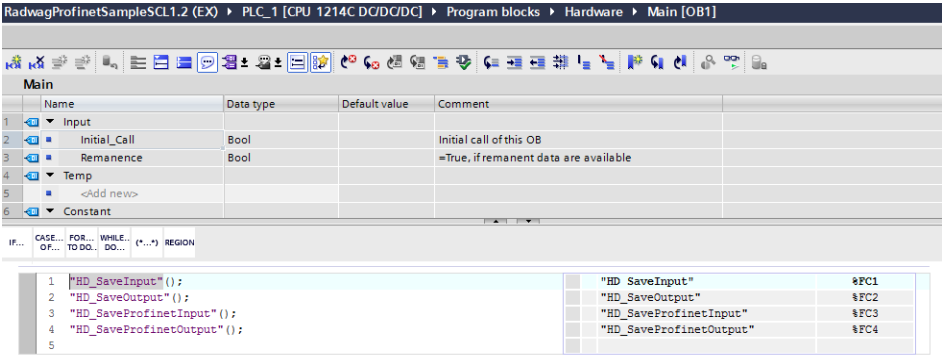
	Name	Data type	Start value	Retain	Accessible f...	Writa...	Visible in ...	Setpoint	Comment
1	Static								
2	komenda	Word	16#02						
3	komenda z parametr...	Word	16#0008						
4	platfroma	Word	16#0001						
5	tara ustaw	Real	2.0						
6	próg LO zapis	Real	1.5						
7	stan wyjść	Word	16#0000						
8	min ustaw	Real	2.1						
9	max ustaw	Real	2.2						
10	numer serii ustaw	DWord	16#0000						
11	operator wybierz	Word	16#0004						
12	towar wybierz	Word	16#0001						
13	kontrahent wybierz	Word	16#01						
14	opakowanie wybierz	Word	16#0004						
15	magazyn źródłowy wy...	Word	16#0						
16	magazyn docelowy w...	Word	16#0						
17	receptura/proces dozo...	Word	16#0						

The functions that rewrite values between physical inputs/outputs of the module may look like this:





Invoke the functions in the main program loop.



Upon compiling and loading the program to the device in the data block, you can read interesting output registers (MONITOR ALL) and save output registers (e.g. by changing START VALUE and LOAD START VALUES AS ACTUAL) of the SLAVE mode.