



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!

Software manual

ITKP-20-01-04-20-EN

PROFINET

Communication Protocol:

MWMH Module

MWLH Module

MWSH Module

APRIL 2020

CONTENTS

- 1. MODULE SETTINGS AND CONFIGURATION..... 4
- 2. DATA STRUCTURE 4
 - 2.1. Input Registers 4
 - 2.2. Output Registers 6
- 3. CONFIGURATION OF PROFINET MODULE IN TIA PORTAL V14 ENVIRONMENT 9
 - 3.1. Import GSD..... 9
 - 3.2. Module Configuration 11
- 4. DIAGNOSTICS APP..... 16

1. MODULE SETTINGS AND CONFIGURATION

Settings of MWMH, MWLH, MWSH modules for communication via PROFINET protocol are configured using MWMH MANAGER software that is to be downloaded from www.radwag.com website. Run the MWMH MANAGER program, go to COMMUNICATION → PERIPHERALS tab, click 'Profinet protocol' entry, next enter the IP address, the subnet mask and the name of a device from the Profinet network. For more information visit www.radwag.com and read user manual of the given device.

2. DATA STRUCTURE

2.1. Input Registers

Input variables:

Variable	Offset	Length [WORD]	Data type
Mass	0	2	float
Tare	4	2	float
Unit	8	1	word
Platform status	10	1	word
LO threshold	12	2	float
Input status	66	1	word
Min	68	2	float
Max	72	2	float
Fast dosing threshold	76	2	float
Slow dosing threshold	80	2	float
Adjustment status	100	1	word
Dosing process status	102	1	word

Platform mass – returns load mass in a current unit.

Platform tare – returns platform tare in an adjustment unit.

Platform unit – determines a current mass unit of the platform.

Register bit	
0	gram [g]
1	kilogram [kg]
2	pound [lb]
3	ounce [oz]
4	carat [ct]
5	Newton [N]

Example:

Read HEX value: 0x02. Binary form:

B1/7	B1/6	B1/5	B1/4	B1/3	B1/2	B1/1	B1/0	B0/7	B0/6	B0/5	B0/4	B0/3	B0/2	B0/1	B0/0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0

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

Platform status – determines state of a given weighing platform.

Register bit	
0	measurement correct (the weighing instrument does not report any error)
1	measurement stable
2	weighing instrument indicates zero
3	weighing instrument tared
4	weighing instrument in II weighing range
5	weighing instrument in III weighing range
6	weighing instrument reports NULL error
7	weighing instrument reports LH error
8	weighing instrument reports FULL error
9	time_triggered_adjustment_required / temperature_triggered_adjustment_required

Example:

Read HEX value: 0x13

B1/7	B1/6	B1/5	B1/4	B1/3	B1/2	B1/1	B1/0	B0/7	B0/6	B0/5	B0/4	B0/3	B0/2	B0/1	B0/0
0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	1

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

LO threshold – returns value of platform's **LO** threshold in an adjustment unit.

Input state – bitmask of platform inputs.

Example:

Read HEX value: 0x02

B1/7	B1/6	B1/5	B1/4	B1/3	B1/2	B1/1	B1/0	B0/7	B0/6	B0/5	B0/4	B0/3	B0/2	B0/1	B0/0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1

Inputs 1 and 2 take HI state

MIN – returns **MIN** threshold value (in a current unit)

MAX – returns **MAX** threshold value (in a current unit).

Fast dosing threshold – returns fast dosing threshold value (rough dosing)

Slow dosing threshold – returns slow dosing threshold value (fine dosing)

Adjustment status

0x00 – adjustment completed correctly

0x01 – process active

0x02 – range exceeded

0x03 – time exceeded

0x04 – process aborted

Dosing process status

0x00 – process disabled

0x01 – taring in progress

0x02 – process activated

0x03 – process inhibited

0x05 – process completed

2.2. Output Registers

Output variables:

Variable	Offset	Length [WORD]	Data type
Command	0	1	word
Command with parameter	2	1	word
Tare	6	2	float
LO threshold	10	2	float
Output state	14	1	word
Min	16	2	float
Max	20	2	float
Fast dosing threshold	24	2	dword
Slow dosing threshold	28	1	word
Adjustment weight mass	48	1	word

Basic command – setting a respective value triggers direct performance of a given task, see the table:


Register bit	Command
0	Zero the platform
1	Tare the platform
5	Dosing start
6	Dosing stop
7	Internal adjustment start
8	Start mass determination
9	Adjustment factor determination
10	Adjustment parameters record (start mass/adjustment factor)

Example:

Record of the register by value 0x02



B1/7	B1/6	B1/5	B1/4	B1/3	B1/2	B1/1	B1/0	B0/7	B0/6	B0/5	B0/4	B0/3	B0/2	B0/1	B0/0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0

Causes platform taring.

	<i>A command is executed once upon detecting that its bit has been set. If the command is to be executed more than once, it is necessary to zero the bit first, and reset it to the required value next.</i>
-----------------------------------------------------------------------------------	---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

Complex command – setting a respective value triggers performance of a given task, see the table:

Register bit	Command
0	Setting tare value for a given platform
1	Setting LO threshold value for a given platform
2	Setting output status
3	Setting MIN threshold
4	Setting MAX threshold
5	Setting fast dosing threshold
6	Setting slow dosing threshold
7	Setting adjustment weight value

	Complex command requires setting the address of a respective parameter (from 2 to 24 – refer to: "Complex command parameters").
	A complex command is executed once upon detecting that its bit has been set. If the command is to be executed more than once, it is necessary to zero the bit first, and reset it to the required value next

Example:

Sending tare of 1.0 value to the scale

Performance of the command requires record of 2 registers:

1. Complex command - value 0x01 – i.e. tare setting.
2. Tare – value 1.0 (0x3F800000).

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

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

Output state – complex command parameter: determination of platform output state.

Example:

Setting high state for output 1 of the platform.

Output mask:

B1/7	B1/6	B1/5	B1/4	B1/3	B1/2	B1/1	B1/0	B0/7	B0/6	B0/5	B0/4	B0/3	B0/2	B0/1	B0/0
0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	1

Upon conversion to HEX the result is 0x01

Performance of the command requires record of 2 registers:

1. Complex command - value 0x02 – i.e. output state setting.
2. Output mask – value 0x01.

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

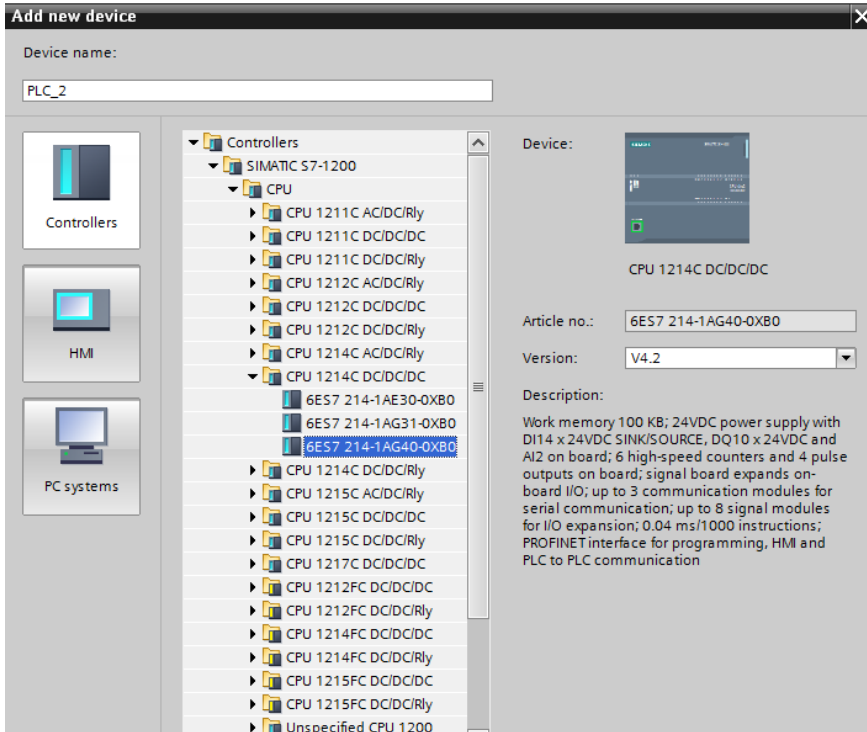
MAX – complex command parameter: MAX threshold value (in the unit set for the active working mode).

Fast dosing threshold – complex command parameter – value of fast dosing (rough dosing) threshold.

Slow dosing threshold – complex command parameter – value of slow dosing (fine dosing) threshold.

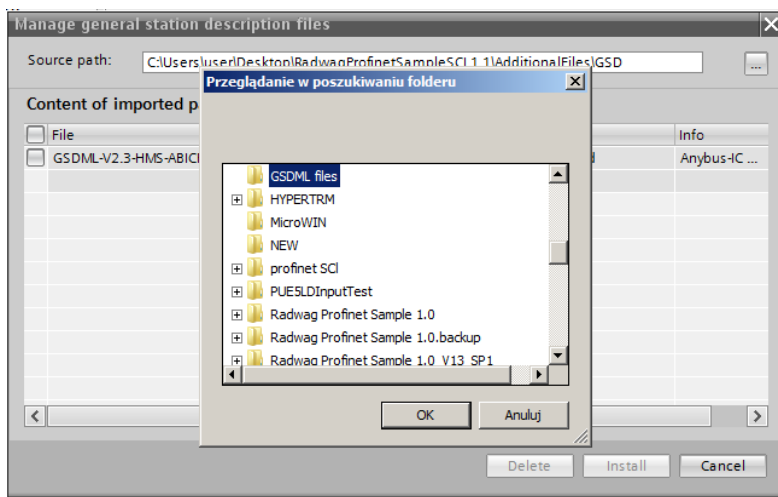
3. CONFIGURATION OF PROFINET MODULE IN TIA PORTAL V14 ENVIRONMENT

Start operation in the environment by setting up a new project, where topology of PROFINET network with MASTER controller will be specified, in this case the MASTER controller is SIEMENS controller of S7-1200 series.

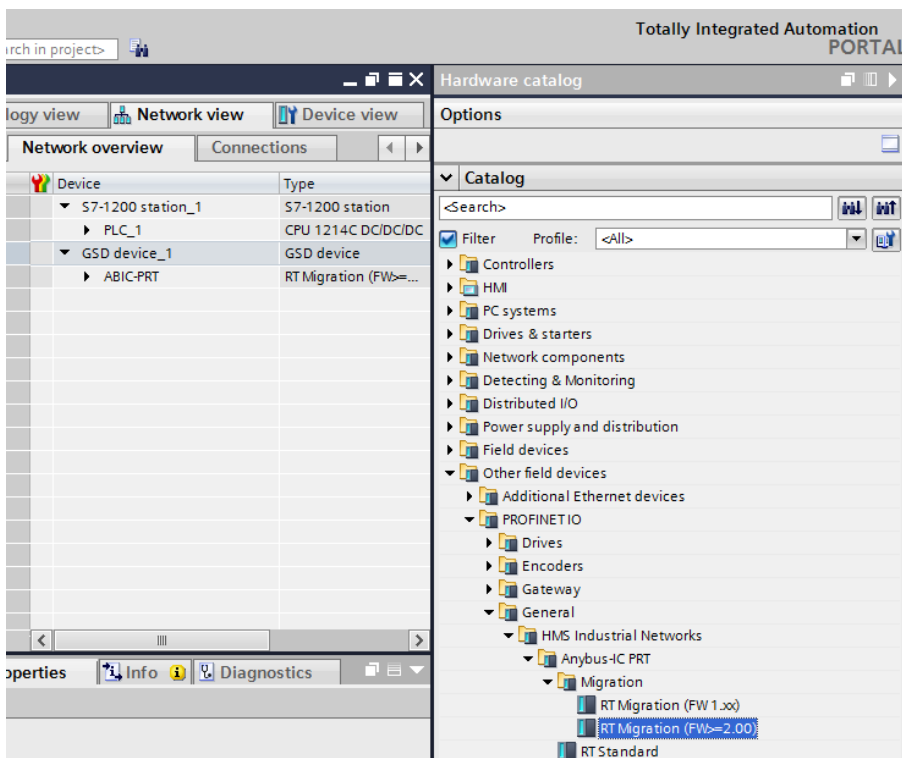


3.1. Import GSD

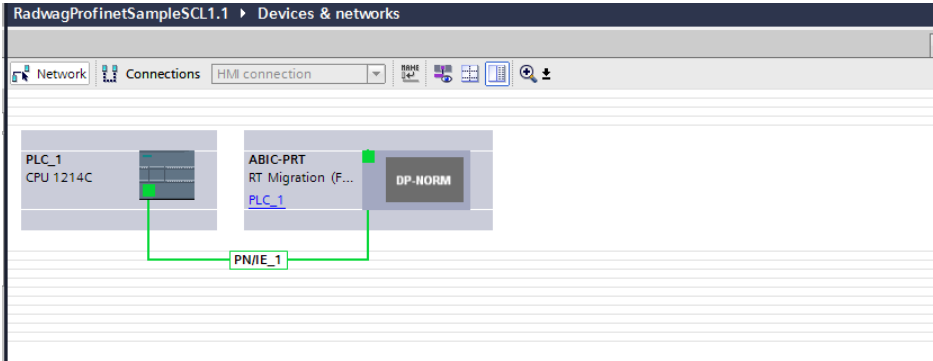
Using a delivered configuration file (GSD), add a new device to the environment. Open OPTIONS tab, next click MANAGE GENERAL STATION DESCRIPTION FILES (GSD) entry and select access path to the GSD file.



With the file successfully added, the ABIC-PRT module is displayed on the list of the devices.



Now it is possible to make a network comprising a MASTER controller and a newly added SLAVE module.



3.2. Module Configuration

At this stage it is necessary to make network comprised of a MASTER controller and a SLAVE device (the scale). With the power supply on, you can use ACCESSIBLE DEVICES function to search for a given device in the environment. The list should display both the MASTER and the SLAVE.

Accessible devices

Type of the PG/PC interface:

PG/PC interface:

Accessible nodes of the selected interface:

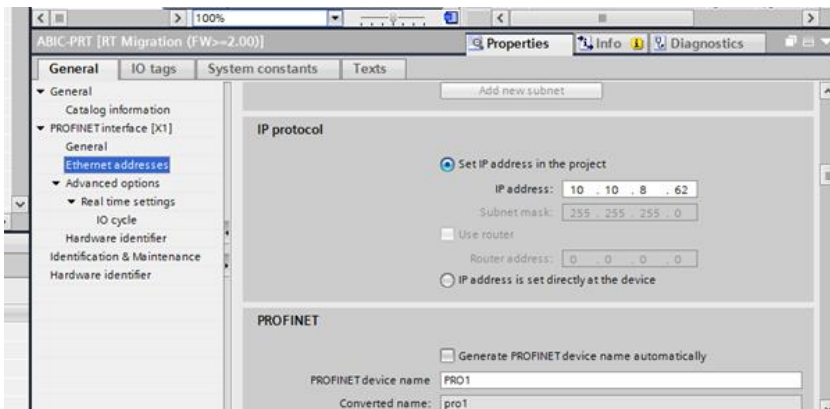
Device	Device type	Interface type	Address	MAC address
Accessible device	S7-PC	ISO	---	00-16-76-25-13-51
pro2	RT Migration (FW 1.xx)	PN/IE	10.10.8.64	00-30-11-0D-EE-17
plc_1	CPU 1214C DC/DC/DC	PN/IE	10.10.8.244	28-63-36-9C-D1-12

☒ Flash LED

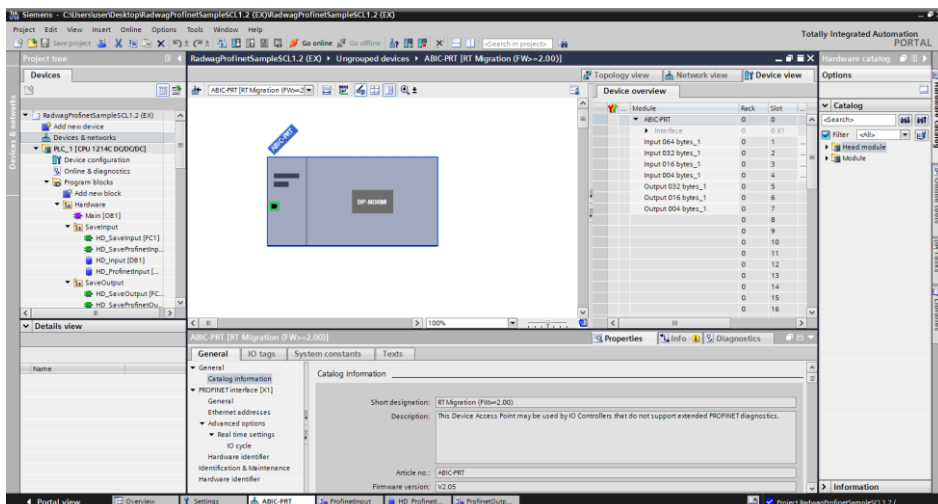
Online status information: ☐ Display only error messages

- Found accessible device Accessible device [00-16-76-25-13-51]
- Scan completed. 3 devices found.
- Retrieving device information...
- Scan and information retrieval completed.

Now, specify the IP address and the name for the module in a PROFINET network. To do it, select the module, in PROPERTIES tab find PROFINET INTERFACE submenu, click it and enter the IP address and name. Make sure that these settings are accordant with parameters set in the scale menu. Remember that the SLAVE IP address and MASTER address must come from the same subnet.



Next configure the module. First specify input and output registers size, define their start addresses. From the list of available INPUT and OUTPUT modules select such modules as presented in the picture below. Maximum size of input data is 110 bytes, and Maximum size of input data is 52 bytes. In the project, default start addresses have been used – 68 for the INPUT module and 64 for the OUTPUT module:



Siemens - C:\Users\user\Desktop\RadwagProfiNetSampleSCL1.2 (EX)\RadwagProfiNetSampleSCL1.2 (EX)

Project Edit View Insert Online Options Tools Window Help

Totally Integrated Automation PORTAL

Project tree: RadwagProfiNetSampleSCL1.2 (EX) > Ungrouped devices > ABC-PR1 [RT Migration (FW=2.00)]

Devices & networks: RadwagProfiNetSampleSCL1.2 (EX) > Devices & networks > PLC_1 [CPU 1214C-2 DP] > Device configuration > Online & diagnostics > Program blocks > Add new block > Hardware > Main [D81] > SaveInput > HD_SaveInput [PC1] > HD_SaveProfiNetIn > HD_Input [D81] > HD_ProfiNetInput [L] > SaveOutput > HD_SaveOutput [PC1] > HD_SaveProfiNetOut

Details view: Name

Topology view: ABC-PR1 [RT Migration (FW=2.00)]

Device overview:

Module	Back	Slot
ABC-PR1	0	0
Interface	0	0 x1
Input 004 bytes_1	0	1
Input 002 bytes_1	0	2
Input 016 bytes_1	0	3
Input 004 bytes_1	0	4
Output 002 bytes_1	0	5
Output 016 bytes_1	0	6
Output 004 bytes_1	0	7

Input 004 bytes_1 [Input 004 bytes]

Properties: General

IO addresses:

Input addresses:

Start address: 68
End address: 131
Organization block: (Automatic update)
Process image: Automatic update

Portal view Overview Settings ABC-PR1 HD_ProfiNet HD_ProfiNetOut

Siemens - C:\Users\user\Desktop\RadwagProfiNetSampleSCL1.2 (EX)\RadwagProfiNetSampleSCL1.2 (EX)

Project Edit View Insert Online Options Tools Window Help

Totally Integrated Automation PORTAL

Project tree: RadwagProfiNetSampleSCL1.2 (EX) > Ungrouped devices > ABC-PR1 [RT Migration (FW=2.00)]

Devices & networks: RadwagProfiNetSampleSCL1.2 (EX) > Devices & networks > PLC_1 [CPU 1214C-2 DP] > Device configuration > Online & diagnostics > Program blocks > Add new block > Hardware > Main [D81] > SaveInput > HD_SaveInput [PC1] > HD_SaveProfiNetIn > HD_Input [D81] > HD_ProfiNetInput [L] > SaveOutput > HD_SaveOutput [PC1] > HD_SaveProfiNetOut

Details view: Name

Topology view: ABC-PR1 [RT Migration (FW=2.00)]

Device overview:

Module	Back	Slot
ABC-PR1	0	0
Interface	0	0 x1
Input 004 bytes_1	0	1
Input 002 bytes_1	0	2
Input 016 bytes_1	0	3
Input 004 bytes_1	0	4
Output 002 bytes_1	0	5
Output 016 bytes_1	0	6
Output 004 bytes_1	0	7

Output 002 bytes_1 [Output 002 bytes]

Properties: General

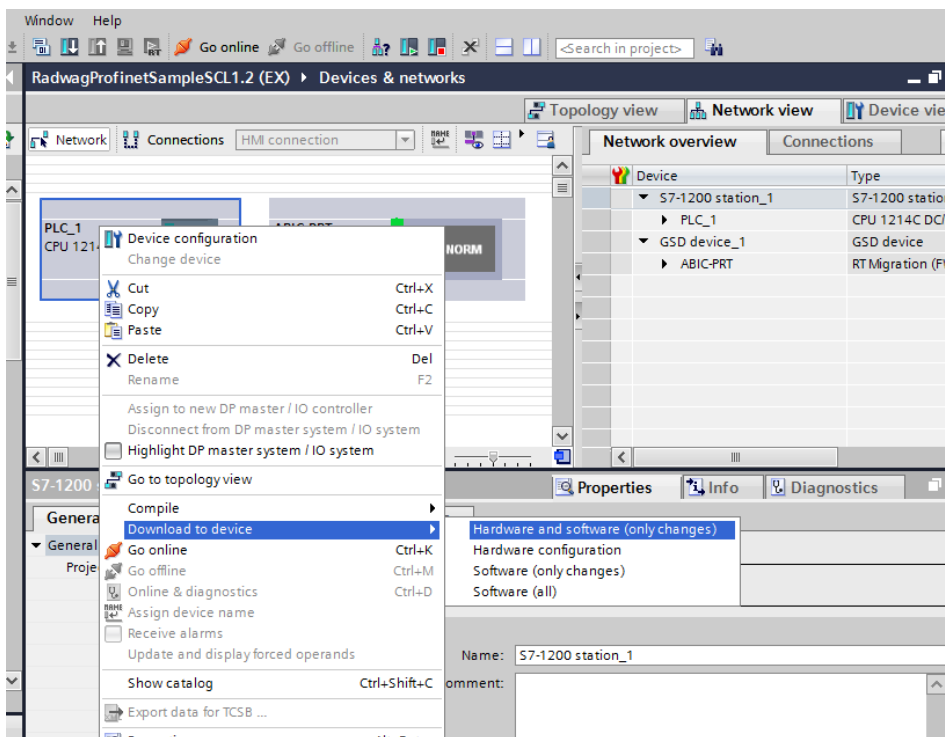
IO addresses:

Output addresses:

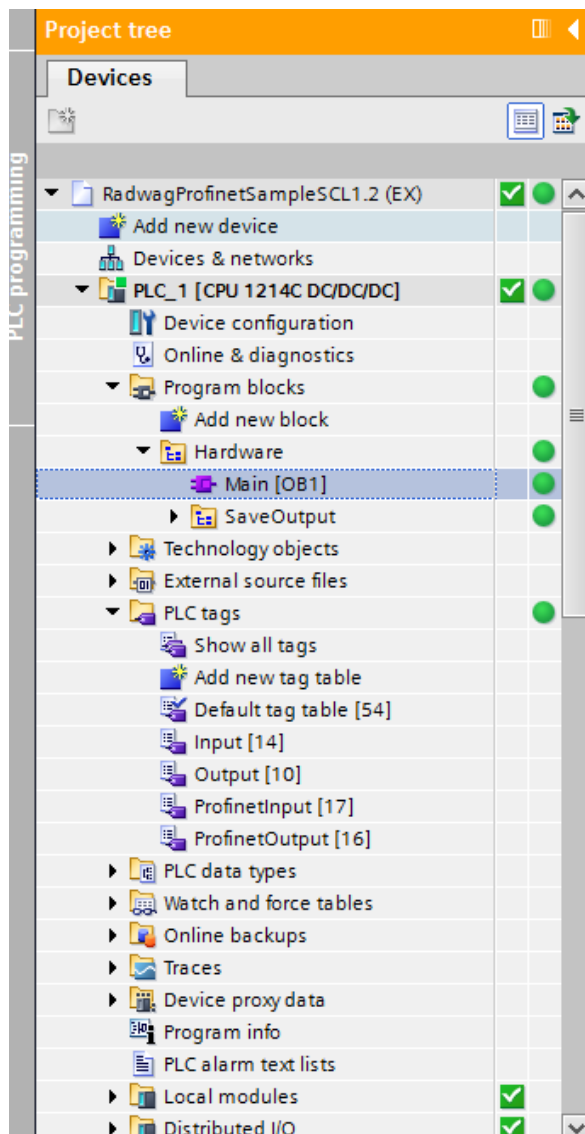
Start address: 64
End address: 95
Organization block: (Automatic update)
Process image: Automatic update

Portal view Overview Settings ABC-PR1 HD_ProfiNet HD_ProfiNetOut

At this stage it is possible to upload the hardware configuration into the controller, and to start data upload.



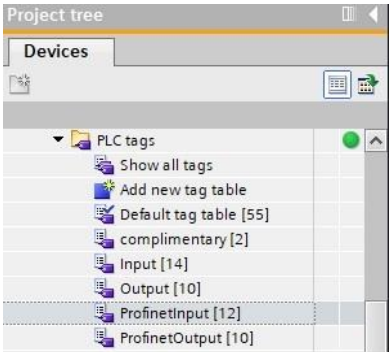
Upon successful compilation and code reading, the MASTER and SLAVE shall establish communication. To check it go online (select 'Go online' entry). The below screenshot presents what you shall see.



Now proceed to the process of program code making.

4. DIAGNOSTICS APP

Start creating the app by defining names of symbolic input and output registers. Use the PLC TAGS directory from the directory tree. For the purpose of this example, the following tables of tags have been created:



INPUT and OUTPUT tables refer to physical inputs/outputs of MASTER controller and are of no importance in case of this application. Input and output registers of PROFINET module have been specified in ProfinetInput and ProfinetOutput tables.

The below screenshots present symbolic names and addresses:

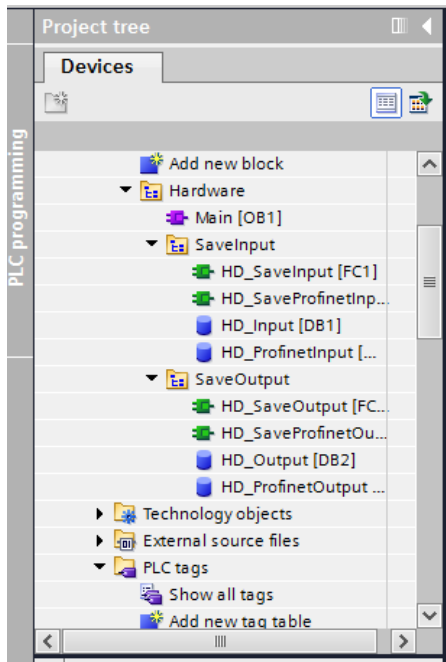
RadwagProfinetHRP1.0 ang ▶ PLC_1 [CPU 1214C DC/DC] ▶ PLC tags ▶ ProfinetInput [12]

	Name	Data type	Address	Retain	Acces...	Writa...	Visibl...	Monitor value	Comment
1	mass	Real	%D68		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	-125.3	
2	tare	Real	%D72		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	125.3	
3	unit	Word	%W76		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	16#0001	
4	status	Word	%W78		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	16#020F	
5	LO	Real	%D80		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	0.0	
6	inputs	Word	%W134		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	16#0000	
7	min	Real	%D136		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	0.0	
8	max	Real	%D140		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	0.0	
9	threshold_dose_coarse	Real	%D144		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	500.0	
10	threshold_dose_fine	Real	%D148		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	990.0	
11	calibr_stat	Word	%W168		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	16#0000	
12	dose_stat	Word	%W170		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	16#0000	
13	<Add new>				<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		

RadwagProfinetHRP1.0 ang ▶ PLC_1 [CPU 1214C DC/DC] ▶ PLC tags ▶ ProfinetOutput [10]

	Name	Data type	Address	Retain	Acces...	Writa...	Visibl...	Monitor value	Comment
1	cammand	Word	%QW64		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	16#0000	
2	complex command	Word	%QW66		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	16#0000	
3	set tare	DWord	%QD70		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	16#0000_0001	
4	set lo	Real	%QD74		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	4.0	
5	outputs	Word	%QW78		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	16#0002	
6	set min	Real	%QD80		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	200.0	
7	set max	Real	%QD84		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	122.34	
8	set threshold_dose_coarse	Real	%QD88		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	444.44	
9	set threshold_dose_fine	Real	%QD92		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	888.88	
10	set_calibr_mass	Real	%QD112		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	2000.0	
11	<Add new>				<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		

So as not to work directly on the physical inputs/outputs of the module it is recommended to create data blocks comprising representations of these registers, and functions "copying" values between them. For this purpose HARDWARE group is created in the PROGRAM BLOCKS directory, the blocks are defined as below:



HD_OUTPUT and HD_INPUT blocks refer to physical MASTER's inputs/outputs and are of no importance in case of this project. HD_ProfinetOutput and HD_ProfinetInput blocks represent input/output registers of the scale's PROFINET module. See the screenshots below:

RadwagProfinetHRP1.0.ang ▶ PLC_1 [CPU 1214C DC/DC] ▶ Program blocks ▶ Hardware ▶ SaveInput ▶ HD_ProfinetInput [DB3]

HD_ProfinetInput									
	Name	Data type	Start value	Retain	Accessible f...	Writa...	Visible in ...	Setpoint	Comment
1	Static								
2	mass	Real	0.0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
3	tare	Real	0.0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
4	unit	Word	16#0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
5	status	Word	16#0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
6	lo	Real	0.0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
7	inputs	Word	16#0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
8	min	Real	16#0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
9	max	Real	0.0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
10	threshold_dose_coarse	Real	0.0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
11	threshold_dose_fine	Real	0.0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
12	calibr_status	Word	16#0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
13	dose_status	Word	16#0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	

RadwagProfinetHRP1.0 ang ▶ 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

	Name	Data type	Start value	Retain	Accessible f...	Write...	Visible in ...	Setpoint	Comment
1	Static								
2	command	Word	16#0000						
3	complex command	Word	16#0000						
4	set tare	Real	1.0						
5	set lo	Real	4.0						
6	outputs	Word	16#0002						
7	set min	Real	200.0						
8	set max	Real	122.34						
9	threshold_dose_coarse	Real	444.44						
10	threshold_dose_fine	Real	888.88						
11	calibr_mass	Real	2000.0						

Functions copying values between the physical inputs/outputs of the module may look as presented below:

RadwagProfinetHRP1.0 ang ▶ PLC_1 [CPU 1214C DC/DC/DC] ▶ Program blocks ▶ Hardware ▶ SaveInput ▶ HD_SaveProfinetInput [FC3]

	Name	Data type	Default value	Comment
1	Input			
2	<Add new>			
3	Output			

```

IF... CASE... FOR... WHILE... (V...) REGION
OF... TO DO... DO...

1 "HD_ProfinetInput".mass := "mass";
2 "HD_ProfinetInput".tare := "tare";
3 "HD_ProfinetInput".unit := "unit";
4 "HD_ProfinetInput".status := "status";
5 "HD_ProfinetInput".lo := "LO";
6 "HD_ProfinetInput".inputs := "inputs";
7 "HD_ProfinetInput".min := "min";
8 "HD_ProfinetInput".max := "max";
9 "HD_ProfinetInput".threshold_dose_coarse := "threshold_dose_coarse";
10 "HD_ProfinetInput".threshold_dose_fine := "threshold_dose_fine";
11 "HD_ProfinetInput".calibr_status := "calibr_stat";
12 "HD_ProfinetInput".dose_status := "dose_stat";
13
14

```

▶ "HD_ProfinetInput" \$DB
 ▶ "HD_ProfinetInput" \$DB
 ▶ "HD_ProfinetInput" \$DB
 ▶ "HD_ProfinetInput" \$DB
 ▶ "HD_ProfinetInput" \$DB
 ▶ "HD_ProfinetInput" \$DB
 ▶ "HD_ProfinetInput" \$DB
 ▶ "HD_ProfinetInput" \$DB
 ▶ "HD_ProfinetInput" \$DB
 ▶ "HD_ProfinetInput" \$DB

RadwagProfinetHRP1.0 ang ▶ PLC_1 [CPU 1214C DC/DC/DC] ▶ Program blocks ▶ Hardware ▶ SaveOutput ▶ HD_SaveProfinetOutput [FC4]

	Name	Data type	Default value	Comment
1	Input			
2	<Add new>			
3	Output			

```

IF... CASE... FOR... WHILE... (V...) REGION
OF... TO DO... DO...

1 "command" := "HD_ProfinetOutput".command;
2 "complex command" := "HD_ProfinetOutput".complex command;
3 "set tare" := "HD_ProfinetOutput".set tare;
4 "set lo" := "HD_ProfinetOutput".set lo;
5 "outputs" := "HD_ProfinetOutput".outputs;
6 "set min" := "HD_ProfinetOutput".set min;
7 "set max" := "HD_ProfinetOutput".set max;
8 "set threshold_dose_coarse" := "HD_ProfinetOutput".threshold_dose_coarse;
9 "set threshold_dose_fine" := "HD_ProfinetOutput".threshold_dose_fine;
10 "set_calibr_mass" := "HD_ProfinetOutput".calibr_mass;
11
12
13

```

▶ "command" %QW64
 ▶ "complex command" %QW66
 ▶ "set tare" %QD70
 ▶ "set lo" %QD74
 ▶ "outputs" %QW78
 ▶ "set min" %QD80
 ▶ "set max" %QD84
 ▶ "set threshold_dose_coa..." %QD88
 ▶ "set threshold_dose_fine" %QD92
 ▶ "set_calibr_mass" %QD112

Now all that needs to be done is enforcing respective functions in the main loop of the program.

RadwagProfinetSampleSCL1.2 (EX) ▶ PLC_1 [CPU 1214C DC/DC/DC] ▶ Program blocks ▶ Hardware ▶ Main [OB1]

	Name	Data type	Default value	Comment
1	Input			
2	Initial_Call	Bool		Initial call of this OB
3	Remanence	Bool		=True, if remanent data are available
4	Temp			
5	<Add new>			
6	Constant			

```

IF... CASE... FOR... WHILE...
OF... TO DO... DO... (*..*) REGION

1 "HD_SaveInput"();
2 "HD_SaveOutput"();
3 "HD_SaveProfinetInput"();
4 "HD_SaveProfinetOutput"();
5

```

"HD_SaveInput"	%FC1	
"HD_SaveOutput"	%FC2	
"HD_SaveProfinetInput"	%FC3	
"HD_SaveProfinetOutput"	%FC4	

Upon compilation and upload of the program to the controller in data block it is possible to read input registers (MONITOR ALL) and to record output registers (e.g. by change of START VALUE and LOAD START VALUES AS ACTUAL) of the SLAVE module.

Example of record of the register by 0x01 value (platform zeroing):

RadwagProfinetHRP1.0 ang ▶ PLC_1 [CPU 1214C DC/DC/DC] ▶ Program blocks ▶ Hardware ▶ Main [OB1]

	Name	Data type	Default value	Comment
1	Input			
2	Initial_Call	Bool		Initial call of this OB
3	Remanence	Bool		=True, if remanent data are available

```

IF... CASE... FOR... WHILE...
OF... TO DO... DO... (*..*) REGION

Network 2:
Comment
1 "imp" := "Input0_0" AND "memo";
2 "memo" := NOT "Input0_0";
3
4 IF ("Input0_0" = true) THEN
5   "HD_ProfinetOutput".command := 16#0001;
6 ELSE
7   "HD_ProfinetOutput".command := 16#0000;
8
9 END_IF;
10

```

"imp"	%M100.0	
"memo"	%M100.1	
"Input0_0"	%I0.0	
"HD_ProfinetOutput"	%DB4	
"HD_ProfinetOutput".com...		
"HD_ProfinetOutput"		

