PROFINET

Communication Protocol:
MW-01-A Mass Converter

SOFTWARE MANUAL

ITKP-40-01-12-21-EN

CONTENTS

1. MASS CONVERTER SETTINGS AND CONFIGURATION

To configure MW-01-A mass converter settings for communication via PROFINET protocol, run **MwManager** PC software and go to **<Parameters / Communication / Additional modules>**. For detailed description of configuration read **MWManager** user manual.

2. DATA STRUCTURE

All registers are 2-byte type (WORD). Floating point data (such as mass and tare) are stored in two consecutive registers and are FLOAT-type data. If the first register consists of two AB bytes and the other of two CD bytes, then FLOAT value is HEX ABCD,

2.1. Input Address

Input variables:

Variable	Offset	Length [WORD]	Data type
Platform mass	0	2	float
Platform tare	4	2	float
Platform unit	8	1	word
Platform status	10	1	word
Platform LO threshold	12	2	float
Process status	64	1	word
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

2.2.1 Input Registers

Platform mass – returns platform net mass in a current unit.

Example:
The read register with offset 0 has a hex value of 0x43E28000, after converting to float we get 453.0 which is the current indication of the load mass.

Platform tare – returns platform tare in an adjustment unit.

Platform unit – determines a current (displayed) net mass unit of the platform.

Unit bits	
0	Gram [g]
1	Kilogram [kg]
2	Carat [ct]
3	Pound [lb]
4	Ounce [oz]
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 mass unit is kilogram [kg].

Platform status – determines status of a weighing platform.

Status bits	
0	Measurement correct (the scale does not report any error)
1	Measurement stable
2	Scale indicates zero
3	Scale tared
4	Scale in II weighing range
5	Scale in III weighing range
6	Scale reports NULL error
7	Scale reports LH error
8	Scale reports FULL error

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

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

Process status – determines status of the dosing\formulation process:

HEX value	Description
0x00	Process disabled
0x01	Process started
0x02	Process aborted
0x03	Process finished

Input state – bitmask of mass converter inputs. Three least significant bits represent the mass converter inputs state.

Example:

Read HEX value: 0x0005

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	1	0	1

Inputs number 1 and 3 of the weighing terminal take HI state.

MIN – returns **MIN** threshold value in an adjustment unit.

MAX – returns **MAX** threshold value in an adjustment unit.

Fast dosing threshold - returns fast dosing threshold value in an adjustment unit.

Slow dosing threshold - returns slow dosing threshold value in an adjustment unit.

Adjustment status – determines adjustment process status:

HEX value	Description
0x00	Process finished correctly
0x01	Start mass/adjustment coefficient determination in progress
0x02	Range exceeded
0x03	Time exceeded
0x04	Process aborted
0x05	Awaiting for data

2.2. Output Address

Output variables list:

Variable	Offset	Length [WORD]	Data type
Basic command	0	1	word
Command with parameter	2	1	word
Tare	6	2	float
LO threshold	10	2	float
Outputs state	14	1	word
Min	16	2	float
Max	20	2	float
Fast dosing threshold	24	2	float
Slow dosing threshold	28	2	float

2.2.1 Output Registers

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

Bit No.	Operation
0	Zero platform
1	Tare platform
5	Process start
6	Process stop

Example:

Writing the register with 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

This causes scale taring.

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.

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

Bit No.	Operation
0	Setting tare value for a platform
1	Setting LO threshold value for a platform
2	Setting outputs state
3	Setting MIN threshold value
4	Setting MAX threshold value
5	Setting fast dosing threshold
6	Setting slow dosing threshold

	***Complex command requires setting a respective parameter (offset from 6 to 28 – refer to output registers table)***
	***A command with a parameter is executed once when its bit setting is detected. 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 writing of 2 registers:

offset 2 – command with a parameter - value 0x01 – i.e. tare setting,

offset 6 – tare value in float format - 1.0.

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: state of mass converter outputs

Example:

Setting high state for output 1 and 3 of the mass converter.

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	0	0	0	0	1	0	1

Upon conversion to HEX the result is 0x05

Performance of the command requires record of 2 registers:

offset 2 – command with parameter - value 0x04 – i.e. output state record,

offset 14 – output mask 0x05.

As a result, outputs number 1 and 3 take high state.

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

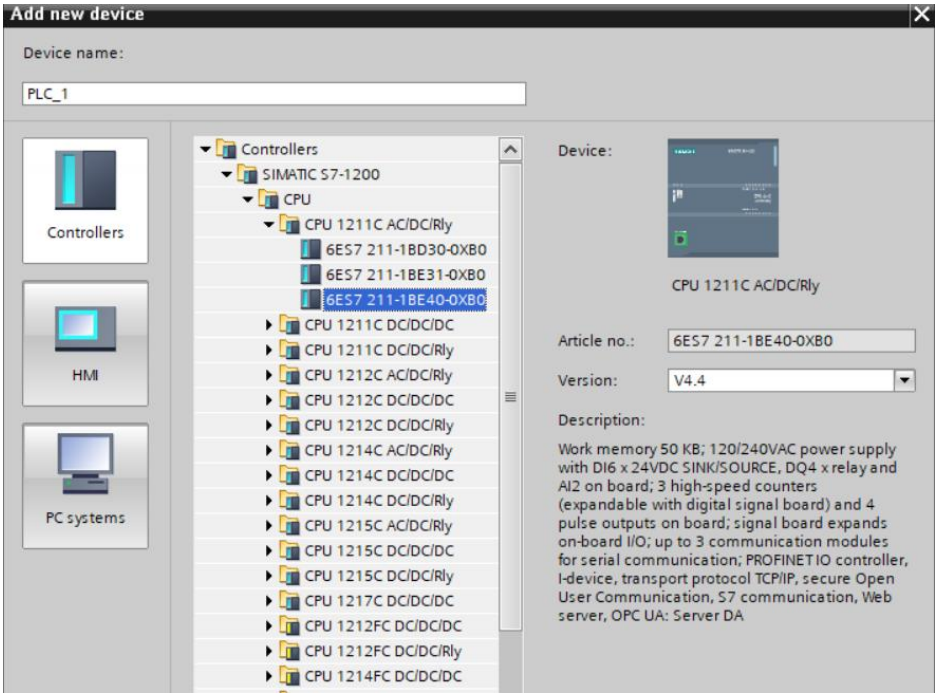
MAX – complex command parameter: MAX threshold value (in an adjustment unit).

Fast dosing threshold - complex command parameter: fast dosing threshold value (in adjustment unit).

Slow dosing threshold - complex command parameter: fine dosing threshold value (in adjustment unit).

3. CONFIGURATION OF THE PROFINET MODULE IN TIA PORTAL V16 ENVIRONMENT

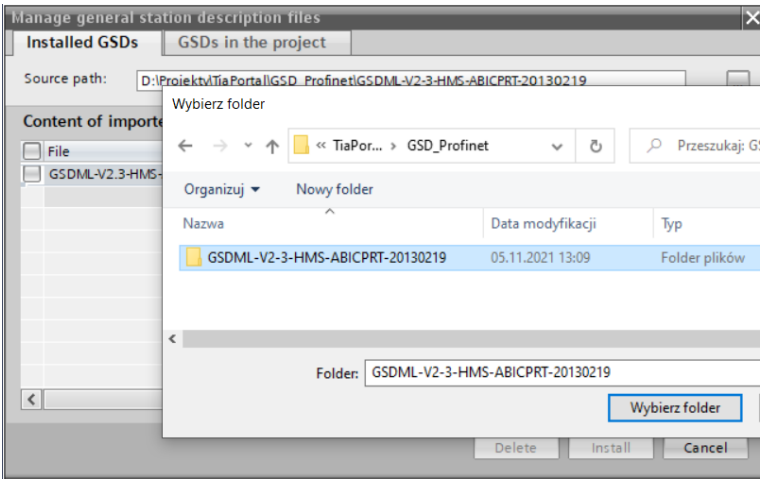
Start operation in the environment by setting up a new project, where topology of PROFINET network with MASTER controller will be specified, in this case the MASTER controller is SIEMENS controller of S7-1200 series.



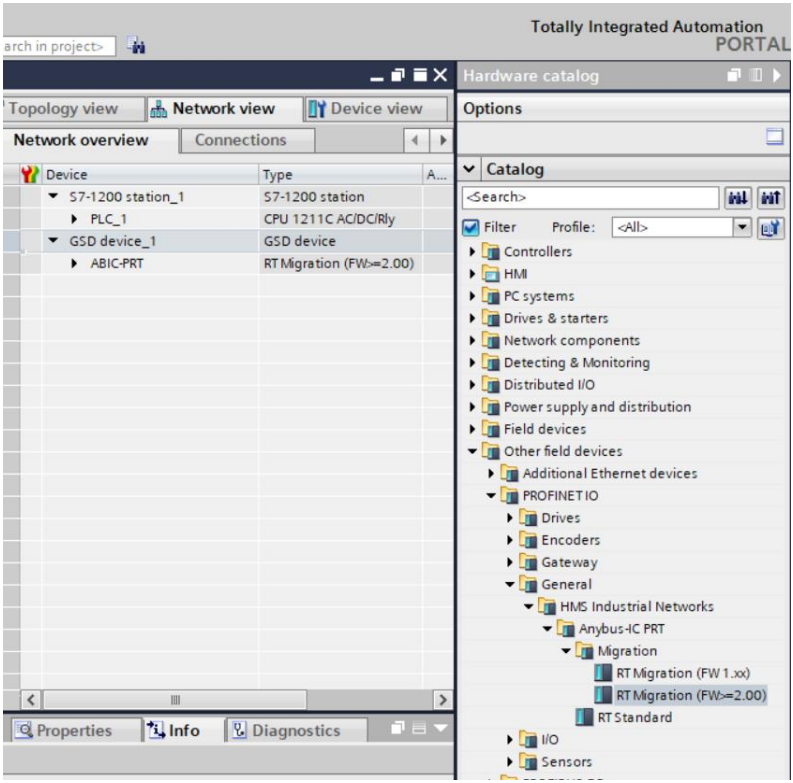
3.1. Import GSD

Using a delivered configuration file (GSD), add a new device to the environment.

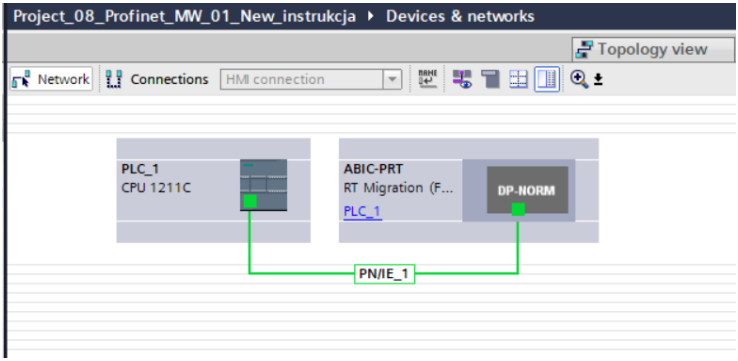
Open OPTIONS tab, next click MANAGE GENERAL STATION DESCRIPTION FILES (GSD) entry and select access path to the GSD file.



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

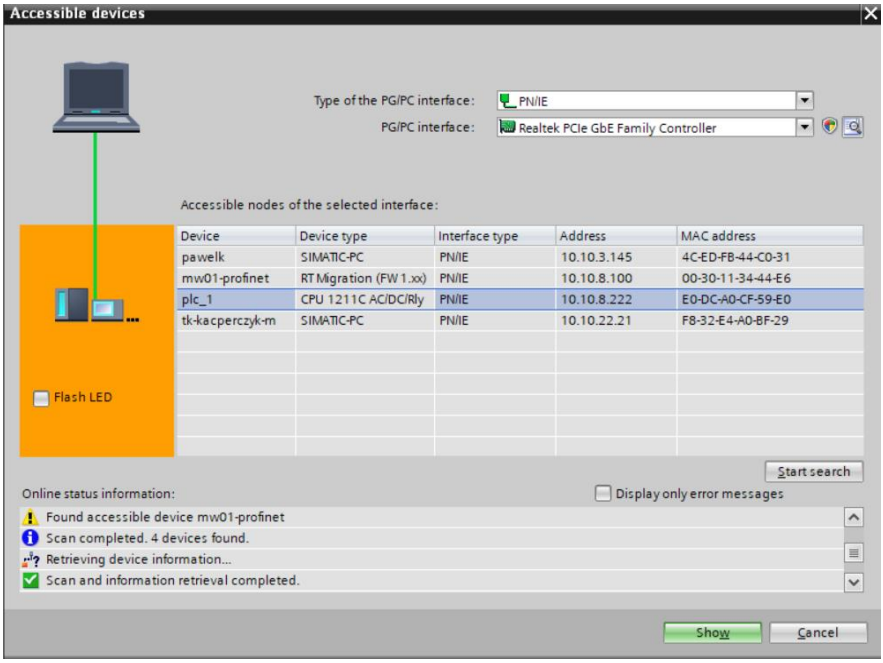


Now, it is possible to make a network comprising a MASTER controller and a newly added SLAVE module.

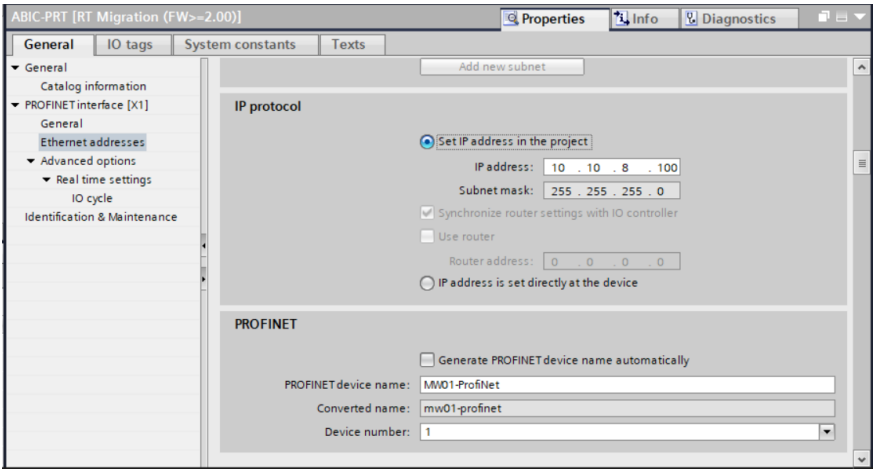


3.2. Module Configuration

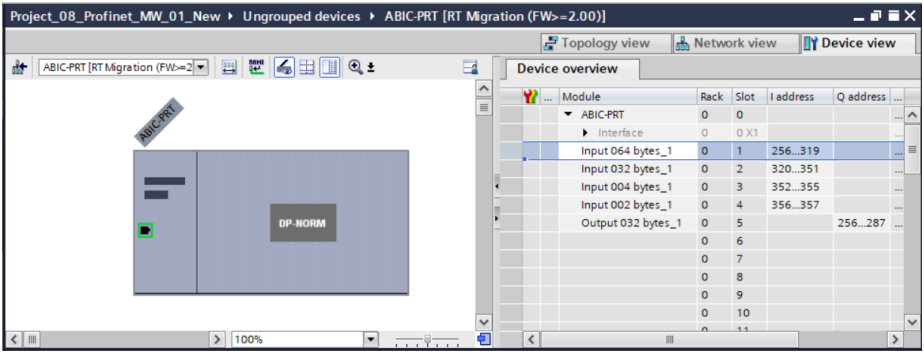
At this stage, it is necessary to create a network consisting of MASTER controller and SLAVE device (MW-01-A mass converter). With the power supply on, you can use ACCESSIBLE DEVICES function to search for a given device in the environment. The list should display both the MASTER and the SLAVE.

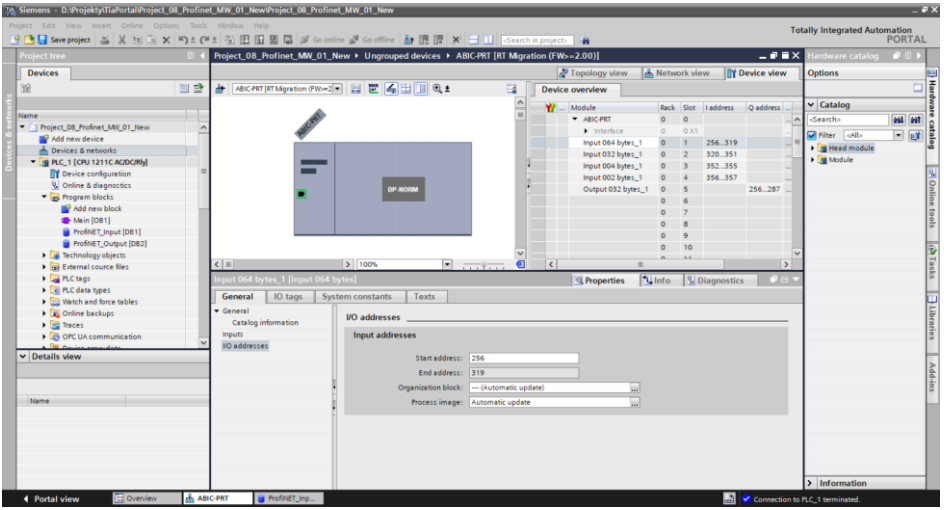


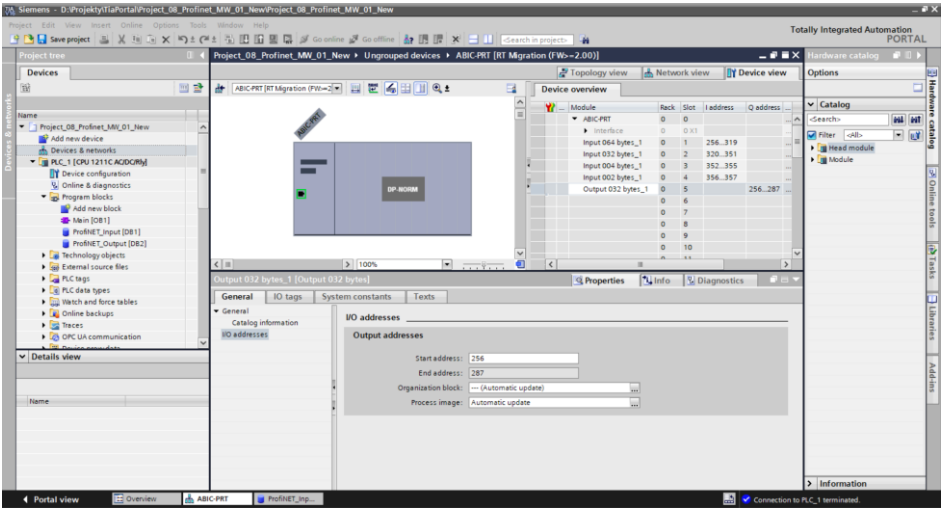
Next, specify the IP address of the module and its name in PROFINET network. To do it, select the module, in PROPERTIES tab find PROFINET INTERFACE submenu, click it and enter the IP address and name. Make sure that these settings are accordant with parameters set in the MW-01 mass converter. Remember that the SLAVE IP address and MASTER address must come from the same subnet.



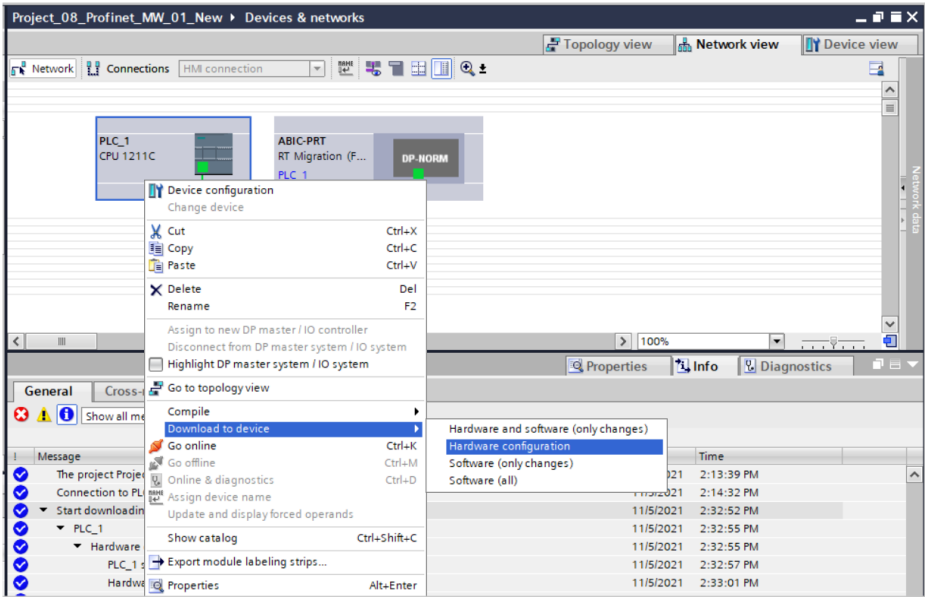
Next, configure the module. First, specify input and output registers size, define their start addresses. From the list of available INPUT and OUTPUT modules select such modules as presented in the picture below. Maximum size of input data is 102 bytes, maximum size of output data is 52 bytes. The project uses the default starting addresses: 256 for INPUT and 256 for OUTPUT with a size of 102 bytes for input data and 32 bytes for output data, as presented below.



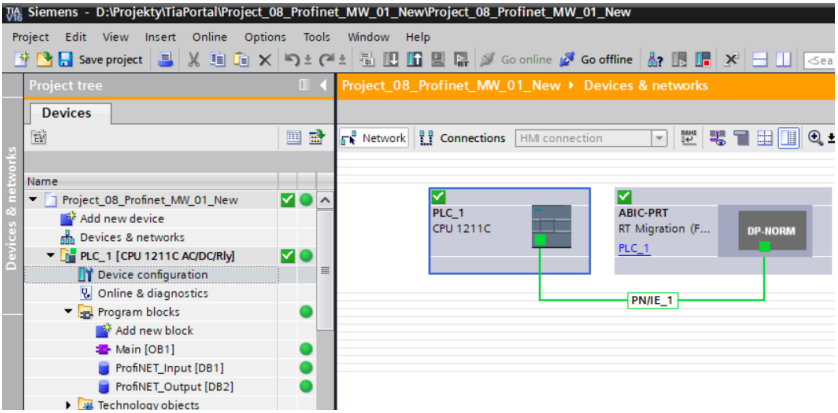




At this stage, it is possible to upload the hardware configuration into the controller.



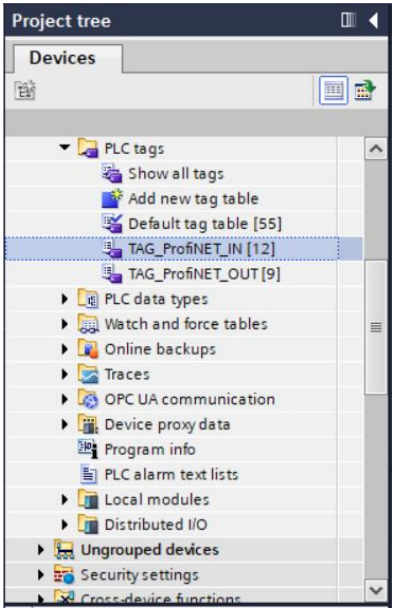
Upon successful compilation and code reading, the MASTER and SLAVE shall establish communication. It can be verified via ONLINE connection. The result should be as presented below:



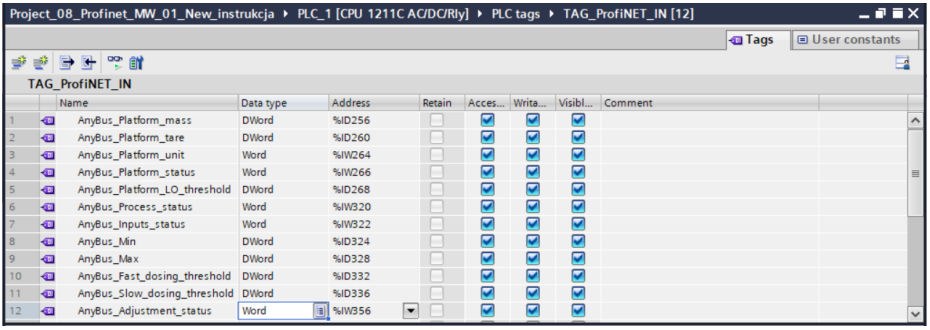
Now, proceed to the process of program code making.

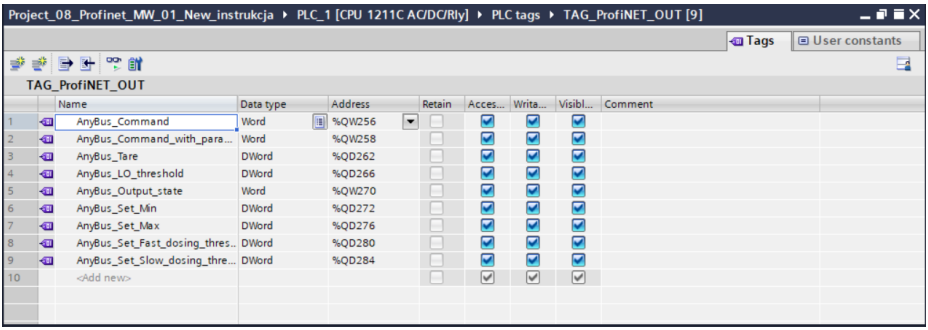
4. DIAGNOSTICS APP

Start creating the app by defining names of symbolic input and output registers. For this purpose, use a branch of the PLC TAGS tree. For the purpose of this example, tag tables have been created as presented below:

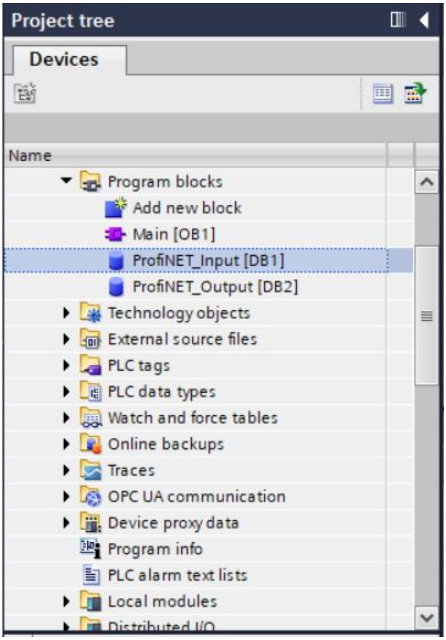


PROFINET input and output registers are specified in TAG_ProfiNET_IN and TAG_ProfiNET_OUT tables. The below screenshots present symbolic names and addresses:

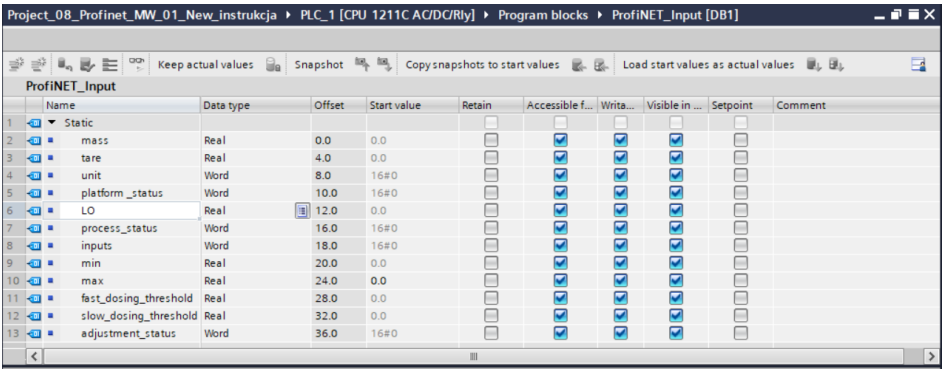


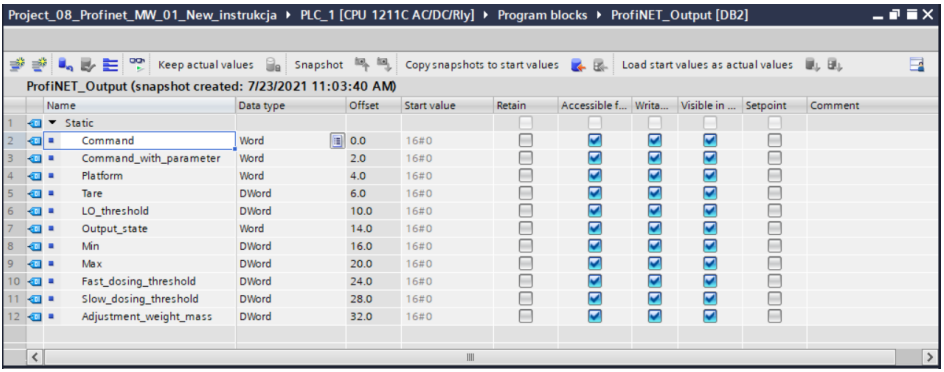


So as not to work directly on the physical inputs/outputs of the module, it is recommended to create data blocks comprising representations of these registers, and “copy” values between them. For this purpose, define data blocks as follows:

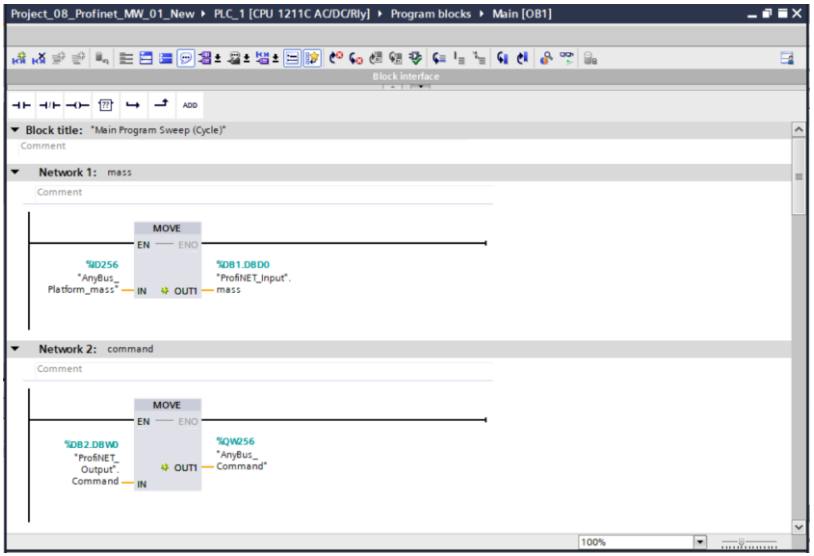


ProfiNET_Input and ProfiNET_Output blocks represent input/output registers of the MW-01-A mass converter's PROFINET module. See the screenshots below:





A MOVE instruction, for example, can be used to rewrite values between the physical inputs/outputs of the module and the registers in the data blocks:



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 changing START VALUE and LOAD START VALUES AS ACTUAL) of the SLAVE module.

RADWAG WAGI ELEKTRONICZNE
ZAAWANSOWANE TECHNOLOGIE WAGOWE