

PROFINET

Protokół komunikacji:

Miernik wagowy PUE HX7

INSTRUKCJA OPROGRAMOWANIA

ITKP-23-02-04-23-PL



KWIECIEŃ 2023

SPIS TREŚCI

1. KONFIGURACJA USTAWIEŃ MIERNIKA WAGOWEGO	4
2. STRUKTURA DANYCH	4
2.1. Adres wejściowy	4
2.2. Opis rejestrów wejściowych	5
2.3. Adres wyjściowy	7
2.4. Opis rejestrów wyjściowych	7
3. KONFIGURACJA MODUŁU PROFINET W ŚRODOWISKU TIA PORTAL V14	11
3.1. Import GSD.....	11
3.2. Konfiguracja modułu.....	13
4. APLIKACJA DIAGNOSTYCZNA.....	17

1. KONFIGURACJA USTAWIENÍ MIERNIKA WAGOWEGO

Konfiguracji ustawień miernika do komunikacji z wykorzystaniem protokołu PROFINET dokonujemy w podmenu <SETUP / Moduły dodatkowe / Moduł anybus>. Konfiguracja ustawień jest szczegółowo opisana w instrukcji „miernik PUE HX7 - Instrukcja oprogramowania”.

2. STRUKTURA DANYCH

2.1. Adres wejściowy

Wykaz zmiennych wejściowych:

Zmienna	Offset	Długość [WORD]	Typ danych
Masa platformy 1	0	2	float
Tara platformy 1	4	2	float
Jednostka platformy 1	8	1	word
Status platformy 1	10	1	word
Próg Lo platformy 1	12	2	float
Masa platformy 2	16	2	float
Tara platformy 2	20	2	float
Jednostka platformy 2	24	1	word
Status platformy 2	26	1	word
Próg Lo platformy 2	28	2	float
Status procesu (Stop, Start)	64	1	word
Stan wejść	66	1	word
Min	68	2	float
Max	72	2	float
Próg dozowania szybkiego	76	2	float
Próg dozowania wolnego	80	2	float
Numer serii	84	2	dword
Operator	88	1	word
Towar	90	1	word
Kontrahent	92	1	word
Opakowanie	94	1	word
Receptura	100	1	word
Proces dozowania	102	1	word

2.2. Opis rejestrów wejściowych

Masa platformy – zwraca wartość masy danej platformy w jednostce aktualnej.

Tara platformy – zwraca wartość tary danej platformy w jednostce kalibracyjnej.

Jednostka platformy – określa aktualną (wyświetlaną) jednostkę masy danej platformy.

Bity jednostki	
0	gram [g]
1	kilogram [kg]
2	karat [ct]
3	funt [lb]
4	uncja [oz]
5	Newton [N]

Przykład:

Wartość odczytana HEX 0x02. Postać binarna:

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

Jednostką wagi jest kilogram [kg].

Status platformy – określa stan danej platformy wagowej.

Bity statusu	
0	Pomiar prawidłowy (waga nie zgłasza błędu)
1	Pomiar stabilny
2	Waga jest w zerze
3	Waga jest wytarowana
4	Waga jest w drugim zakresie
5	Waga jest w trzecim zakresie
6	Waga zgłasza błąd NULL
7	Waga zgłasza błąd LH
8	Waga zgłasza błąd FULL

Przykład:

Odczytana wartość HEX: 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

Waga nie zgłasza błędu, pomiar stabilny w drugim zakresie.

Próg LO – zwraca wartość progu **LO** w jednostce kalibracyjnej danej platformy.

Status procesu – określa status procesu dozowania lub recepturowania:

0x00 – proces nieaktywny

0x01 – proces uruchomiony

0x02 – proces przerwany

0x03 – proces zakończony

Stan wejść – maska bitowa wejść miernika. Pierwsze 4 najmłodsze bity reprezentują wejścia terminala wagowego.

Przykład:

Odczytana wartość HEX: 0x000B

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	1	0	1	1

Wejścia numer 1,2 i 3 terminala wagowego znajdują się w stanie wysokim.

MIN – zwraca wartość ustawionego progu **MIN** w jednostce aktualnej.

MAX – zwraca wartość ustawionego progu **MAX** w jednostce aktualnej.

Próg dozowania szybkiego – zwraca wartość ustawionej wartości masy progu dozowania szybkiego w procesie dozowania.

Próg dozowania wolnego - zwraca wartość ustawionej wartości masy progu dozowania wolnego w procesie dozowania.

Numer serii – zwraca wartość numeru serii. Akceptowane są tylko wartości numeryczne! Wszystkie inne znaki są pomijane.

Operator – zwraca wartość kodu zalogowanego operatora.

Towar – zwraca wartość kodu wybranego towaru.

Kontrahent – zwraca wartość kodu wybranego kontrahenta.

Opakowanie – zwraca wartość kodu wybranego opakowania.

Receptura – zwraca wartość kodu wybranej receptury.

Proces dozowania – zwraca wartość kodu wybranego procesu dozowania.

2.3. Adres wyjściowy

Wykaz zmiennych wyjściowych:


Zmienna	Offset	Długość [WORD]	Typ danych
Komenda	0	1	word
Komenda z parametrem	2	1	word
Platforma	4	1	word
Tara	6	2	float
Próg LO	10	2	float
Stan wyjść	14	1	word
Min	16	2	float
Max	20	2	float
Numer serii	32	2	dword
Operator	36	1	word
Towar	38	1	word
Kontrahent	40	1	word
Opakowanie	42	1	word
Receptura	48	1	word
Proces dozowania	50	1	word

2.4. Opis rejestrów wyjściowych

Komenda podstawowa – zapisanie rejestru odpowiednią wartością spowoduje wywołanie następujących akcji:

Numer bitu	Akcja
0	Zeruj platformę
1	Taruj platformę
2	Wyczyść statystyki
3	Zapisz/Drukuj
4	Start

5	Awaria (Stop bez potwierdzenia)
6	Taruj/Zeruj platformę
7	Blokuj klawiaturę
8	Odblokuj klawiaturę

	<p><i>Komenda wykonywana jest jednorazowo, po wykryciu ustawienia danego jej bitu. Jeżeli konieczne jest ponowne wykonanie komendy z ustawionym tym samym bitem, należy go najpierw wyzerować a następnie ustawić na żadaną wartość ponownie.</i></p>
---	--

Przykład:



Zapisanie rejestru wartością 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

Spowoduje wytarowanie wagi.

Komenda złożona – ustawienie odpowiedniej wartości realizuje zadanie, zgodnie z tabelą:

Numer bitu	Akcja
0	Ustawienie wartości tary dla danej platformy
1	Ustawienie wartości progu LO dla danej platformy
2	Ustawienie numeru serii
3	Ustawienie stanu wyjść
4	Wybór operatora
5	Wybór produktu
6	Wybór opakowania
7	Ustawienie wartości progu MIN
8	Wybór kontrahenta
11	Wybór procesu dozowania
12	Ustawienie wartości progu MAX

	<i>Komenda złożona wymaga ustawienia odpowiedniego parametru (offset od 4 do 50 – patrz tabela rejestrów wyjściowych).</i>
	<i>Komenda z parametrem wykonywana jest jednorazowo, po wykryciu ustawienia danego jej bitu. Jeżeli konieczne jest ponowne wykonanie komendy z ustawionym tym samym bitem, należy go najpierw wyzerować a następnie ustawić na żądaną wartość ponownie.</i>

Przykład:

Wysłanie do wagi tary o wartości 1.0 dla 1-szej platformy.

Wykonanie komendy wymaga zapisania 3 rejestrów:

offset 2 – komenda z parametrem - wartość 0x01 – czyli ustawienie tary.

offset 4 – numer platformy wagowej, do której chcemy przypisać tarę- wartość 0x01 dla pierwszej platformy.

offset 6 – wartość tary w formacie float - 1.0.

Platforma – parametr komendy złożonej: numer platformy wagowej (1 lub 2).

Tara – parametr komendy złożonej: wartość tary (w jednostce kalibracyjnej).

Próg LO – parametr komendy złożonej: wartość progu LO (w jednostce kalibracyjnej).

Stan wyjść – parametr komendy złożonej: określający stan wyjść miernika wagowego i modułu komunikacyjnego.

Przykład:

Ustawienie w stan wysoki wyjść nr 1 i 3 terminala wagowego.

Maska wyjść będzie miała postać:

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

Po konwersji na HEX otrzymamy 0x05

Wykonanie komendy wymaga zapisania 2 rejestrów:

offset 2 – komenda z parametrem - wartość 0x08 – czyli zapis stanu wyjść.

offset 14 – maska wyjść 0x05

W efekcie wyjścia numer 1 i 3 zostaną ustawione w stan wysoki.

MIN – parametr komendy złożonej: wartość progu MIN (w jednostce aktualnie używanego modu pracy).

MAX – parametr komendy złożonej: wartość progu MAX (w jednostce aktualnie używanego modu pracy).

Numer serii – parametr komendy złożonej: wartość numeru serii. Akceptowane są tylko wartości numeryczne! Wszystkie inne znaki są pomijane.

Operator – parametr komendy złożonej: kod operatora (tylko numeryczny).

Towar – parametr komendy złożonej: kod towaru (tylko numeryczny).

Kontrahent – parametr komendy złożonej: kod kontrahenta (tylko numeryczny).

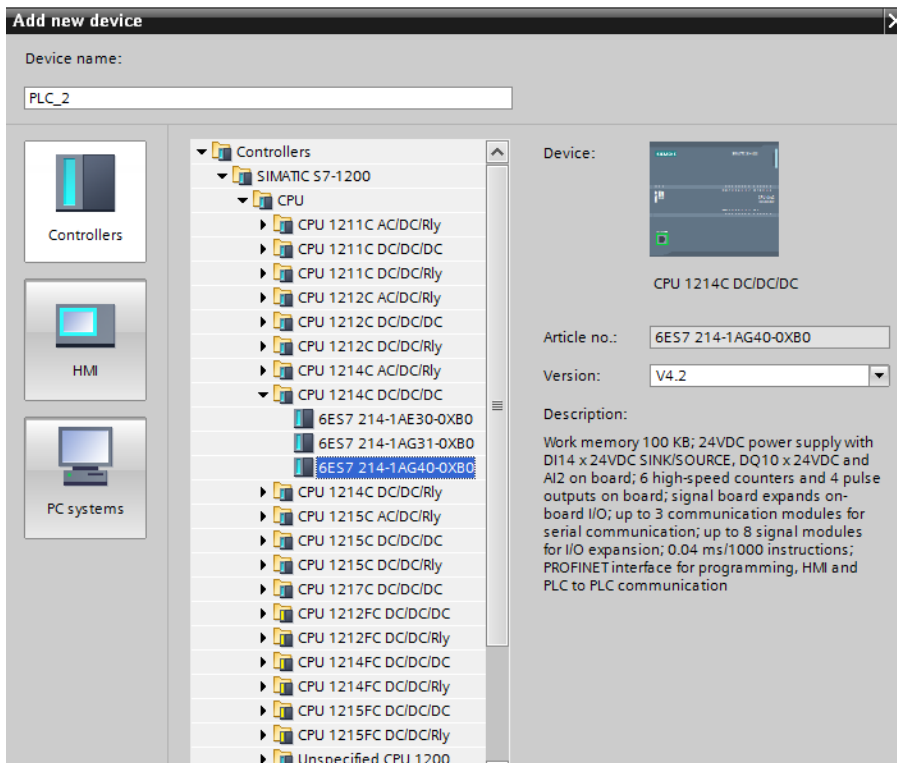
Opakowanie – parametr komendy złożonej: kod opakowania (tylko numeryczny)

Receptura – parametr komendy złożonej: kod receptury (tylko numeryczny).

Proces dozowania – parametr komendy złożonej: kod procesu dozowania (tylko numeryczny).

3. KONFIGURACJA MODUŁU PROFINET W ŚRODOWISKU TIA PORTAL V14

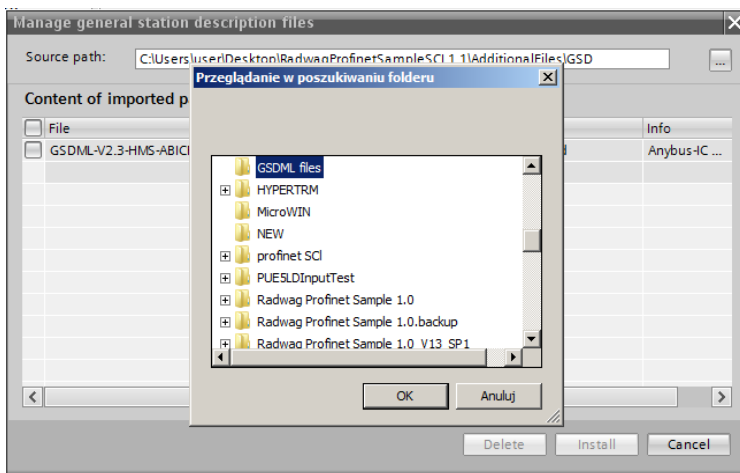
Pracę w środowisku należy rozpocząć od założenia nowego projektu, w którym określona zostanie topologia sieci PROFINET ze sterownikiem MASTER, którym w tym przykładzie będzie sterownik serii S7-1200 firmy SIEMENS.



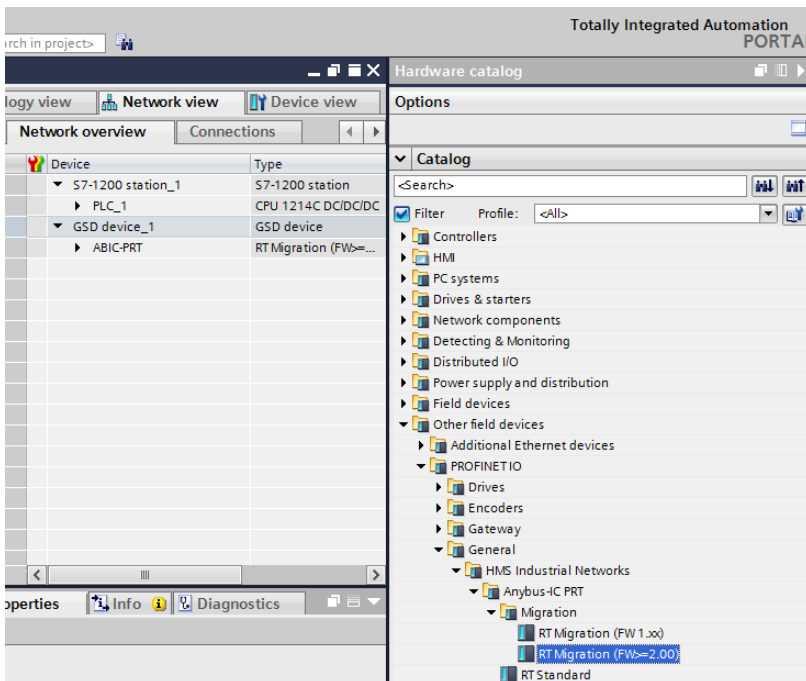
3.1. Import GSD

Korzystając z dołączonego pliku konfiguracyjnego GSD należy dodać nowe urządzenie w środowisku.

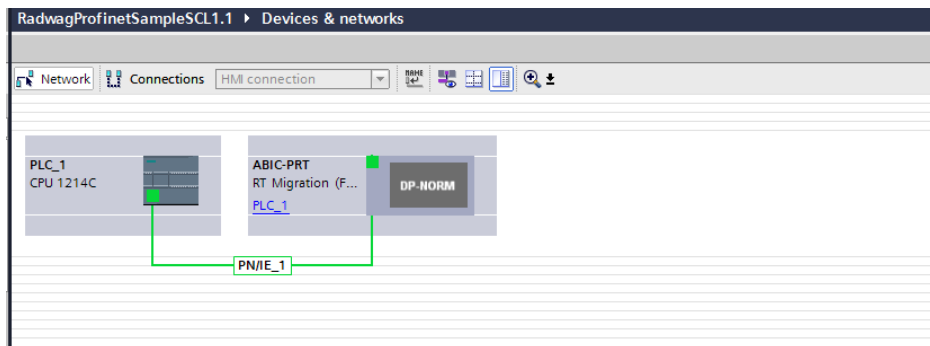
W tym celu należy użyć zakładki OPTIONS a następnie MANAGE GENERAL STATION DESCRIPTION FILES (GSD) i wskazać ścieżkę dostępu do pliku GSD.



Po pomyślnym dodaniu pliku w liście urządzeń możemy już odnaleźć interesujący nas moduł ABIC-PRT:



Można już utworzyć sieć składającą się z jednego sterownika MASTER oraz dodanego modułu SLAVE:



3.2. Konfiguracja modułu

Na tym etapie należy zbudować sieć złożoną ze sterownika MASTER, urządzenia SLAVE (waga). Po podłączeniu zasilania w środowisku można wyszukać urządzenia korzystając z funkcji ACCESSIBLE DEVICES. W efekcie powinniśmy odnaleźć na liście zarówno MASTER jak i SLAVE:

Type of the PG/PC interface:

PG/PC interface:

Accessible nodes of the selected interface:

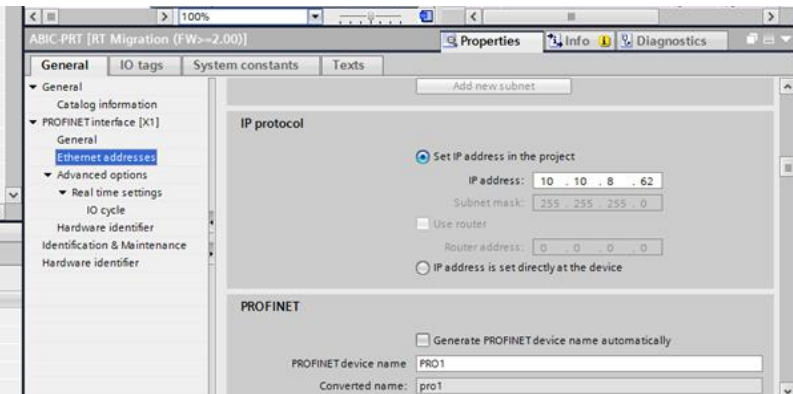
Device	Device type	Interface type	Address	MAC address
Accessible device	S7-PC	ISO	—	00-16-76-25-13-51
pro2	RT Migration (FW 1.xx)	PN/E	10.10.8.64	00-30-11-0D-EE-17
plc_1	CPU 1214C DC/DC/DC	PN/E	10.10.8.244	28-63-36-9C-D1-12

Flash LED

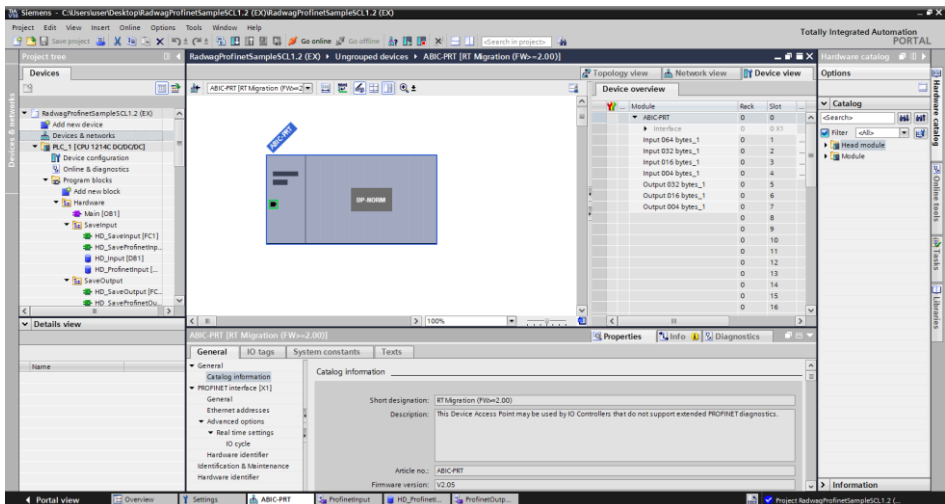
Online status information: Display only error messages

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

W dalszej kolejności należy określić adres IP modułu i jego nazwę w sieci PROFINET. Po zaznaczeniu modułu w zakładce PROPERTIES odnajdujemy podmenu PROFINET INTERFACE gdzie wpisujemy adres IP oraz nadajemy nazwę. Te ustawienia muszą być zgodne z parametrami ustawionymi w menu wagi. Należy pamiętać o tym żeby adres IP SLAVE znajdował się w tej samej podsieci co adres MASTER.



Możemy przejść do konfiguracji modułu. Na wstępie określamy rozmiar rejestrów wejściowych oraz wyjściowych a także definiujemy ich adresy początkowe. W tym celu z listy dostępnych modułów INPUT oraz OUTPUT wybieramy takie jak na zdjęciu poniżej. Maksymalny rozmiar danych wejściowych wynosi 116 bajtów i tyle samo dla danych wyjściowych. W projekcie użyto domyślnych adresów początkowych – 68 dla modułu INPUT i 64 dla OUTPUT:



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 DC) > Device configuration > Device & diagnostics > Program blocks > Add new block > Hardware > Main [DB1] > SaveInput > HD_SaveInput [FC1] > HD_SaveProfinetIn...

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 032 bytes_1	0	5
Output 016 bytes_1	0	6
Output 004 bytes_1	0	7

IO addresses: Input 004 bytes_1 [Input 004 bytes]

General: ID tags, System constants, Texts

IO addresses: Hardware identifier

Input addresses:

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

Portal view | Overview | Settings | ABC-PR1 | ProfinetInput | HD_Profnet... | 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 DC) > Device configuration > Device & diagnostics > Program blocks > Add new block > Hardware > Main [DB1] > SaveOutput > HD_SaveOutput [FC1] > HD_SaveProfinetOu...

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 032 bytes_1	0	5
Output 016 bytes_1	0	6
Output 004 bytes_1	0	7

IO addresses: Output 032 bytes_1 [Output 032 bytes]

General: ID tags, System constants, Texts

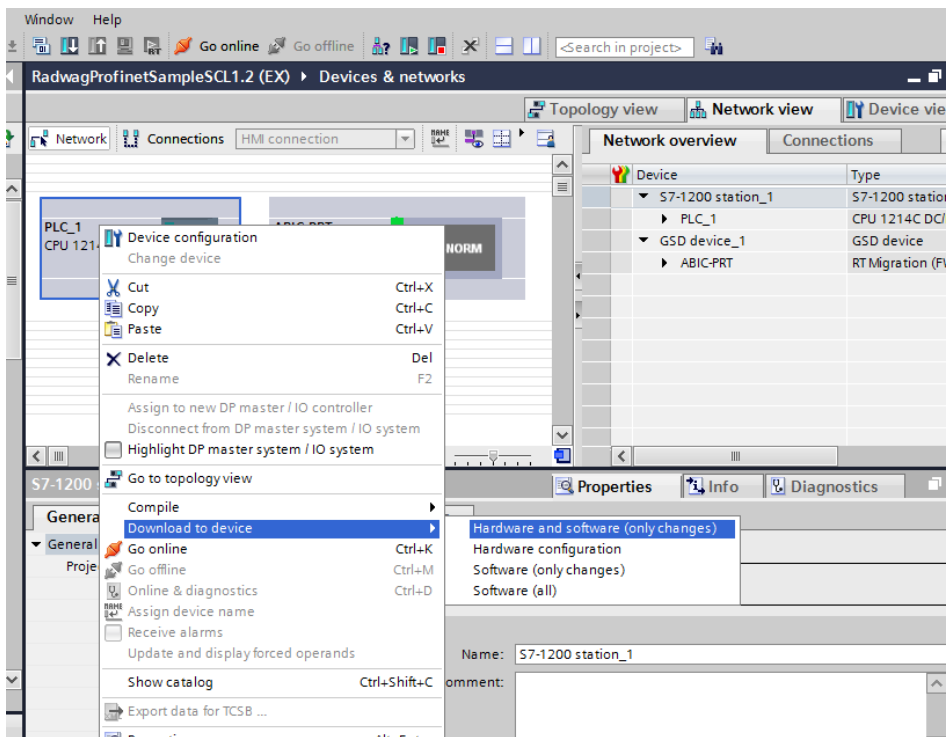
IO addresses: Hardware identifier

Output addresses:

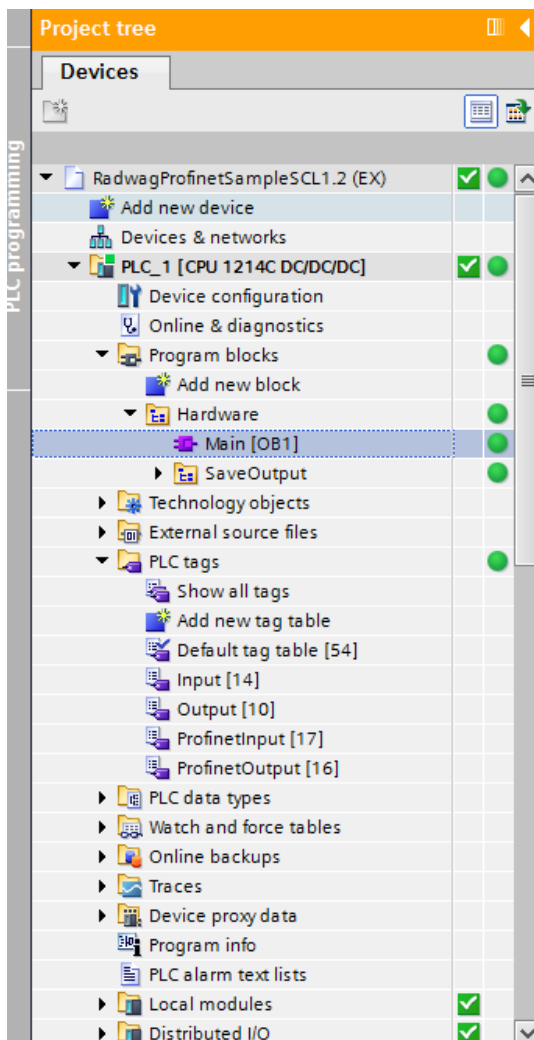
Start address: 64
End address: 95
Organization block: (Automatic update)
Process image: Automatic update

Portal view | Overview | Settings | ABC-PR1 | ProfinetInput | HD_Profnet... | ProfinetOut

Na tym etapie można załadować do sterownika konfigurację sprzętową:



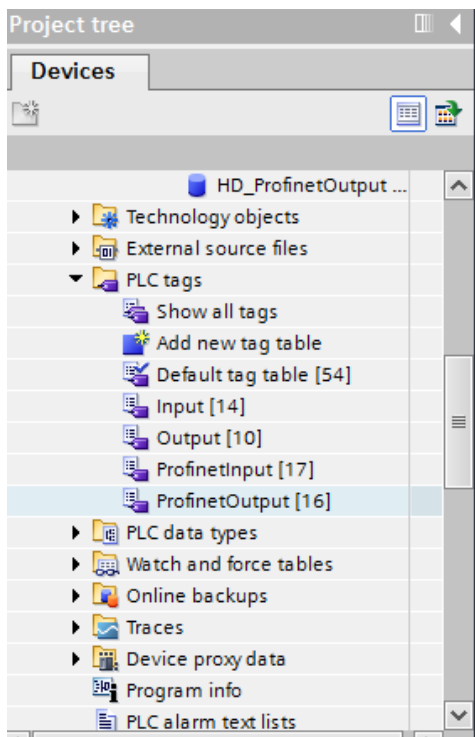
Po pomyślnej kompilacji i wczytaniu kodu MASTER i SLAVE powinny nawiązać połączenie. Można to sprawdzić przechodząc do połączenia ONLINE. Powinniśmy uzyskać wynik jak poniżej.



Dalszym etapem będzie tworzenie kodu programu.

4. APLIKACJA DIAGNOSTYCZNA

Tworzenie aplikacji najlepiej zacząć od zdefiniowania nazw symbolicznych rejestrów wejściowych i wyjściowych. W tym celu korzystamy z gałęzi drzewa projektu o nazwie PLC TAGS. Na potrzeby tego przykładu stworzono tablice tagów jak na rysunku poniżej:



Tablice INPUT i OUTPUT odnoszą się do fizycznych wejść/wyjść sterownika MASTER i nie mają znaczenia w tej aplikacji. Rejestry wejściowe i wyjściowe modułu PROFINET określono w tablicach ProfinetInput oraz ProfinetOutput. Poniższe rysunki prezentują nadane nazwy symboliczne i adresację:

RadwagProfinetHX7 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

HD_ProfinetOutput		Name	Data type	Start value	Retain	Accessible f...	Writa...	Visible in ...	Setpoint	Comment
1	▼	Static								
2	▶	command	Word	16#0000	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
3	▶	complex command	Word	16#0000	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
4	▶	platform	Word	16#01	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
5	▶	set tare	Real	2.0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
6	▶	set lo	Real	4.0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
7	▶	outputs	Word	16#0002	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
8	▶	set min	Real	2.2	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
9	▶	set max	Real	3.3	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
10	▶	set lot number	DWord	444	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
11	▶	set operator	Word	12	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
12	▶	set article	Word	2	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
13	▶	set customer	Word	16#0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
14	▶	set packaging	Word	16#0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
15	▶	set recipe	Word	16#444	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
16	▶	set dosing	Word	16#0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
17	▶	<<Add new>			<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	

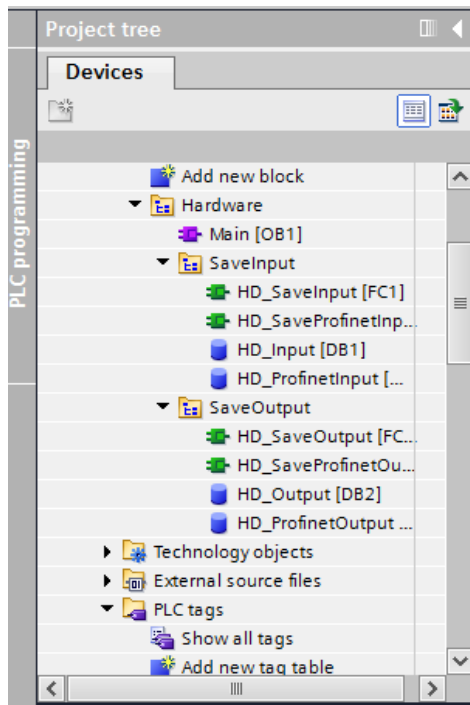
RadwagProfinetHX7 ang > PLC_1 [CPU 1214C DC/DC] > PLC tags > ProfinetInput [20]

Tags User constants

ProfinetInput

	Name	Data type	Address	Retain	Acces...	Writa...	Visibl...	Comment
1	mass_1	Real	%ID68	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
2	tare_1	Real	%ID72	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
3	unit_1	Word	%IW76	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
4	status_1	Word	%IW78	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
5	LO_1	Real	%ID80	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
6	mass_2	Real	%ID84	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
7	tare_2	Real	%ID88	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
8	unit_2	Word	%IW92	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
9	status_2	Word	%IW94	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
10	LO_2	Real	%ID96	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
11	inputs	Word	%IW134	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
12	min	Real	%ID136	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
13	max	Real	%ID140	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
14	lot number	DWord	%ID152	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
15	operator	Word	%IW156	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
16	article	Word	%IW158	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
17	customer	Word	%IW160	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
18	packaging	Word	%IW162	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
19	recipe	Word	%IW168	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
20	dosing	Word	%IW170	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	

Żeby nie pracować bezpośrednio na fizycznych wejściach/wyjściach modułu warto stworzyć bloki danych zawierające reprezentacje tych rejestrów oraz stworzyć funkcje „przepisujące” wartości pomiędzy nimi. W tym celu tworzymy grupę HARDWARE w gałęzi PROGRAM BLOCKS oraz definiujemy bloki danych jak poniżej:



Bloki HD_OUTPUT i HD_INPUT odnoszą się do fizycznych wejść/wyjść MASTER i nie mają znaczenia dla tego projektu. Bloki HD_ProfinetOutput oraz HD_ProfinetInput reprezentują interesujące nas rejestry wejść/wyjść modułu PROFINET wagi. Wyglądają one jak poniżej:

Name	Data type	Start value	Retain	Accessible f...	Writa...	Visible in ...	Setpoint	Comment
1	Static							
2	mass_1	Real	0.0					
3	tare_1	Real	0.0					
4	unit_1	Word	16#0					
5	status_1	Word	16#0					
6	lo_1	Real	0.0					
7	mass_2	Real	0.0					
8	tare_2	Real	0.0					
9	unit_2	Word	16#0					
10	status_2	Word	16#0					
11	lo_2	Real	0.0					
12	inputs	Word	16#0					
13	min	Real	16#0					
14	max	Real	0.0					
15	lot numbet	DWord	16#444					
16	operator	Word	16#0					
17	article	Word	16#0					
18	customer	Word	16#0					
19	packaging	Word	16#0					
20	recipe	Word	16#0					
21	dosing	Word	16#0					

Name	Data type	Start value	Retain	Accessible f...	Writa...	Visible in ...	Setpoint	Comment
1	Static							
2	command	Word	16#0000					
3	complex command	Word	16#0000					
4	platform	Word	16#01					
5	set tare	Real	2.0					
6	set lo	Real	4.0					
7	outputs	Word	16#0002					
8	set min	Real	2.2					
9	set max	Real	3.3					
10	set lot number	DWord	444					
11	set operator	Word	12					
12	set article	Word	2					
13	set customer	Word	16#0					
14	set packaging	Word	16#0					
15	set recipe	Word	16#44					
16	set dosing	Word	16#0					
17	<-Add new>							

Funkcje przepisujące wartości pomiędzy fizycznymi wejściami/wyjściami modułu mogą wyglądać jak poniżej:

RadwagProfinetHX7 ang > PLC_1 [CPU 1214C DC/DC/DC] > Program blocks > Hardware > SaveInput > HD_SaveProfinetInput [FC3]

HD_SaveProfinetInput

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

```

IF... CASE... FOR... WHILE... (*..*) REGION
OF... TO DO... DO...
1 "HD_ProfinetInput".mass_1 := "mass_1";           >> "HD_ProfinetInput"           >>> %DB3
2 "HD_ProfinetInput".tare_1 := "tare_1";           >> "HD_ProfinetInput"           >>> %DB3
3 "HD_ProfinetInput".unit_1 := "unit_1";           >> "HD_ProfinetInput"           >>> %DB3
4 "HD_ProfinetInput".status_1 := "status_1";       >> "HD_ProfinetInput"           >>> %DB3
5 "HD_ProfinetInput".lo_1 := "LO_1";               >> "HD_ProfinetInput"           >>> %DB3
6 "HD_ProfinetInput".mass_2 := "mass_2";           >> "HD_ProfinetInput"           >>> %DB3
7 "HD_ProfinetInput".tare_2 := "tare_2";           >> "HD_ProfinetInput"           >>> %DB3
8 "HD_ProfinetInput".unit_2 := "unit_2";           >> "HD_ProfinetInput"           >>> %DB3
9 "HD_ProfinetInput".status_2 := "status_2";       >> "HD_ProfinetInput"           >>> %DB3
10 "HD_ProfinetInput".lo_2 := "LO_2";              >> "HD_ProfinetInput"           >>> %DB3
11 "HD_ProfinetInput".inputs := "inputs";          >> "HD_ProfinetInput"           >>> %DB3
12 "HD_ProfinetInput".min := "min";                >> "HD_ProfinetInput"           >>> %DB3
13 "HD_ProfinetInput".max := "max";                >> "HD_ProfinetInput"           >>> %DB3
14 "HD_ProfinetInput".lot number := "lot number";   >> "HD_ProfinetInput"           >>> %DB3
15 "HD_ProfinetInput".operator := "operator";       >> "HD_ProfinetInput"           >>> %DB3
16 "HD_ProfinetInput".article := "article";         >> "HD_ProfinetInput"           >>> %DB3
17 "HD_ProfinetInput".customer := "customer";      >> "HD_ProfinetInput"           >>> %DB3
18 "HD_ProfinetInput".packaging := "packaging";    >> "HD_ProfinetInput"           >>> %DB3
19 "HD_ProfinetInput".recipe := "recipe";          >> "HD_ProfinetInput"           >>> %DB3
20 "HD_ProfinetInput".dosing := "dosing";          >> "HD_ProfinetInput"           >>> %DB3
21

```

RadwagProfinetHX7 ang > PLC_1 [CPU 1214C DC/DC/DC] > Program blocks > Hardware > SaveOutput > HD_SaveProfinetOutput [FC4]

HD_SaveProfinetOutput

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

```

IF... CASE... FOR... WHILE... (*..*) REGION
OF... TO DO... DO...
1 "command" := "HD_ProfinetOutput".command;        >> "command"                    >>> %QW64
2 "complex command" := "HD_ProfinetOutput".complex command; >> "complex command"           >>> %QW66
3 "set platform" := "HD_ProfinetOutput".platform;  >> "set platform"               >>> %QW68
4 "set tare" := "HD_ProfinetOutput".set tare";     >> "set tare"                   >>> %QD70
5 "set lo" := "HD_ProfinetOutput".set lo";         >> "set lo"                     >>> %QD74
6 "outputs" := "HD_ProfinetOutput".outputs;       >> "outputs"                    >>> %QW78
7 "set min" := "HD_ProfinetOutput".set min";      >> "set min"                    >>> %QD80
8 "set max" := "HD_ProfinetOutput".set max";      >> "set max"                    >>> %QD84
9 "set lot number" := "HD_ProfinetOutput".set lot number"; >> "set lot number"             >>> %QD96
10 "set operator" := "HD_ProfinetOutput".set operator"; >> "set operator"               >>> %QW100
11 "set article" := "HD_ProfinetOutput".set article"; >> "set article"                 >>> %QW102
12 "set customer" := "HD_ProfinetOutput".set customer"; >> "set customer"               >>> %QW104
13 "set packag %QW104 Word finetOutput".set packaging"; >> "set packaging"              >>> %QW106
14 "set recipe := "HD_ProfinetOutput".set recipe"; >> "set recipe"                 >>> %QW112
15 "set dosing" := "HD_ProfinetOutput".set dosing"; >> "set dosing"                 >>> %QW114

```

Pozostaje w głównej pętli programu wywołać interesujące nas funkcje.

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

Main

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

```

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

Po kompilacji i załadowaniu programu do sterownika w bloku danych możemy odczytać interesujące nas rejestry wejściowe (MONITOR ALL) oraz zapisywać rejestry wyjściowe (np. poprzez zmianę START VALUE i LOAD START VALUES AS ACTUAL) modułu SLAVE.



RADWAG WAGI ELEKTRONICZNE
ZAAWANSOWANE TECHNOLOGIE WAGOWE

